NetSim®

Accelerate Network R & D

Vehicular Ad hoc Networks (VANETs)

A Network Simulation & Emulation Software

By

TETCOS LLP

Contact us at

TETCOS LLP

214, 39th A Cross, 7th Main, 5th Block Jayanagar,

Bangalore - 560 041, Karnataka, INDIA.

Phone: +91 80 26630624

E-Mail: sales@tetcos.com

Visit: www.tetcos.com

Contents

1 Introduction

Connected vehicle (CV) technologies enable a wide range of transportation applications, safety, mobility, and infotainment. While holding tremendous promise, the success of these CV-enabled applications relies heavily on the quality of the underlying information flow. NetSim is a simulation tool designed to model, simulate and analyze this information flow. The vehicular communication architecture in NetSim is based on a combination of the IEEE 1609 standard and IEEE 802.11p standard. The 802.11p standard defines the PHY and MAC layers while IEEE 1609 defines the upper layers.

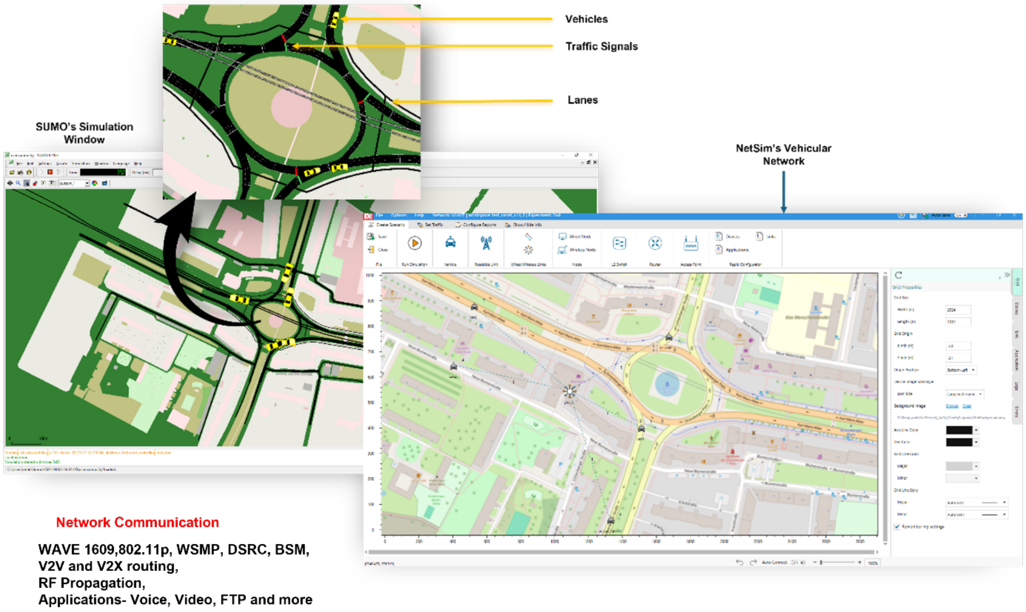


Figure 1-1: *NetSim-SUMO interfacing for VANET simulation. Top left is a SUMO screen shot while bottom right is a NetSim screen shot.*

NetSim's VANET library features:

- IEEE 802.11p PHY operating in the 5.9 GHz band with a channel bandwidth of 10 MHz. 802.11p is an adaptation of the widely used IEEE 802.11a standard previously used in Wi-Fi systems.
- Radio propagation in the PHY layer covering various pathloss, shadowing, and fading models.
- IEEE 802.11p MAC layer. Stations communicate directly outside the context of a BSS.
- IEEE 1609-2, which defines security services for application messages and management messages in WAVE.
- IEEE 1609-3, which defines connection set up and management of WAVE compliant devices.
- IEEE 1609-4, which enables upper layer operational aspects across multiple channels without knowledge of PHY layer parameters.
- DSRC SAE J2735
- BSM packets that are transmitted using WSMP
- A spontaneous Ad hoc network formation between the VANET nodes; layer-3 IP routing can be through DSR, AODV, OLSR or ZRP for non-BSM packets
- Vehicular mobility using in-built mobility models or by interfacing with SUMO software
- Interfacing between SUMO & NetSim via Traffic control interface (TraCI). Automatic import of road network and vehicle mobility from SUMO
- Wide range of output metrics including Delay, Throughput, Error, Retransmission, etc.

- Protocol source C code is provided along with NetSim software

In VANETs, Vehicles and roadside units (RSUs) are the primary communicating nodes, providing each other with (i) Safety information using BSM application and (ii) Infotainment applications. Both types of nodes are Dedicated Short-Range Communications (DSRC) devices. The RSU is a WAVE device usually fixed along the roadside or in dedicated locations such as at junctions or near parking spaces. In NetSim, users can model network traffic flows:

- Between two or more Vehicles, known as V2V
- From vehicles to RSUs (infrastructure), known as V2I
- Between two or more RSUs
- From vehicles or RSUs to remote servers, by connecting RSUs in backhaul to a wired network consisting of switches, routers, and servers for end-to-end simulation.

2 Simulation GUI

2.1 Create Scenario

- Open NetSim and click New Simulation → Vehicular Adhoc Network (Vanet) as shown Figure 2-1.

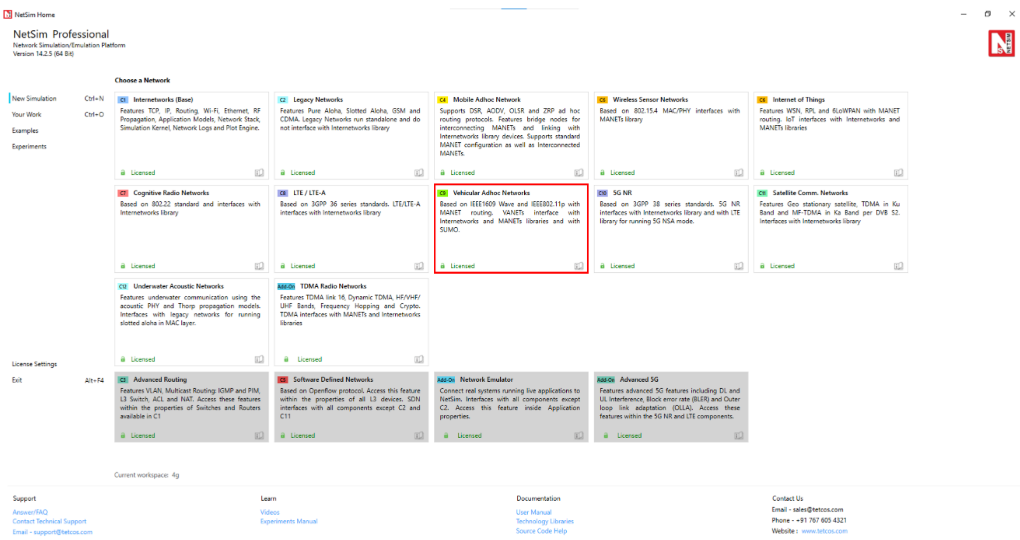


Figure 2-1: *NetSim Home Screen*

- A dialogue box appears as shown below, in that browse the Sumo Configuration File path. The general format of such file is "*.Sumo.cfg".

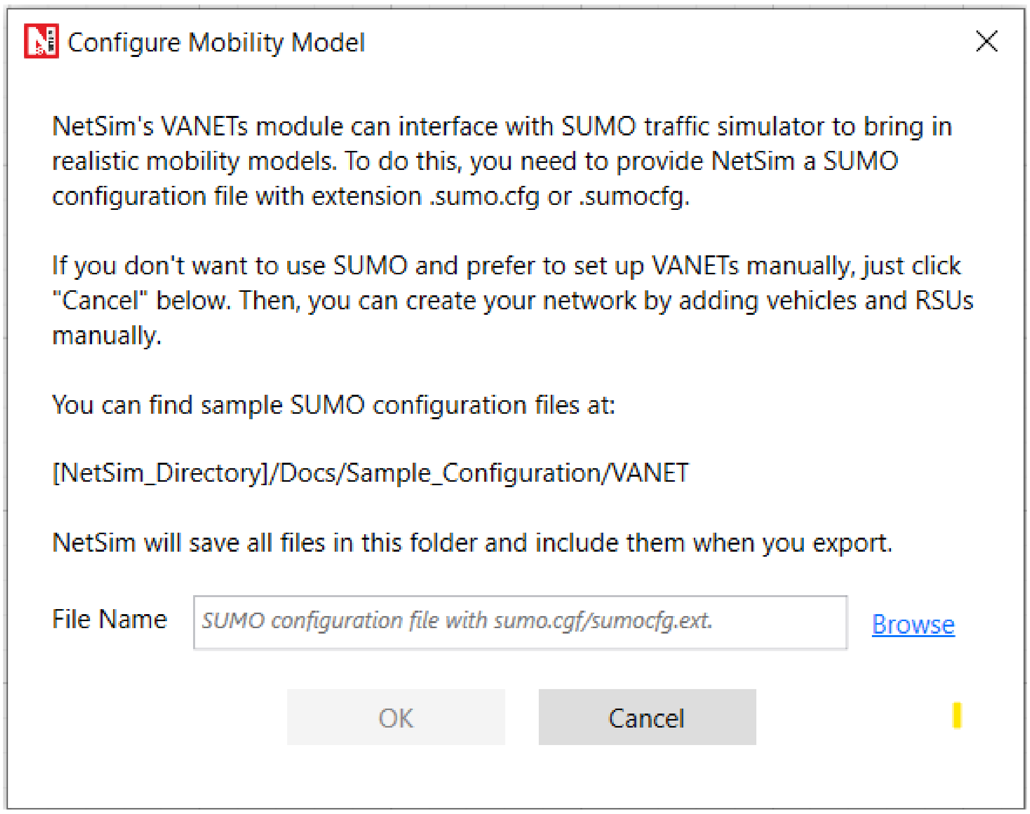


Figure 2-2: *Sumo Configuration File path*

- The NetSim VANET module is designed to interface directly with SUMO.
- A SUMO configuration file is required as an input to NetSim.
- Sample SUMO configuration files are available inside the Config support folder, which is present in each example in the `<NetSim-Installation-Directory>\Docs\Sample Configuration\VANET` folder.
- Users can either use a Sumo configuration file which is provided inside NetSim's installation directory or use a different user specified SUMO configuration file. This .cfg file contains the path of NETWORK file and VEHICLES file.
- Further help on how to create SUMO configuration files is available at `http://sumo.dlr.de/wiki/Networks/Building_Networks_from_own_XML-descriptions`.

After selecting the SUMO configuration file, the scenario opens with nodes placed at their respective starting positions (tracked from SUMO). Roads and traffic lights are also placed exactly as specified in the SUMO configuration file.

2.2 Devices specific to NetSim VANETs Library

- **Vehicle (with one OBU):** In NetSim, a vehicle is a mobile communications device. It is assumed to have one (1) on board unit (OBU) which is a 5-layer device. The OBU can communicate with other OBUs or with RSUs via an Ad hoc link. The OBU is assumed to have one wireless interface and has its own IP and MAC addresses.
- **Roadside Unit (RSU):** In NetSim, an RSU is a fixed communicating device. RSUs are generally termed as infrastructure. Vehicle (OBU) to RSU is termed as V2I communication. The RSU is a 5-layer device that can be connected to a Vehicle or to a Router. RSUs cannot be directly connected to other RSUs. RSUs have one (1) wireless interface and one (1) serial interface, and each interface has its own IP and MAC addresses.

- **Wired node:** A Wired node can be an end-node or for a server. It is a 5-layer device that can be connected to a switch and router. It supports only 1 Ethernet interface and has its own IP and MAC Addresses.
- **Wireless Nodes:** A Wireless node can be an end-node or a server. It is a 5-layer wireless device that can be connected to an Access point. It supports only 1 Wireless interface and has its own IP and MAC Addresses.
- **L2 Switch:** A Switch is a layer-2 device that uses the devices' MAC address to make forwarding decisions. It does not have an IP address.
- **Router:** Router is a layer-3 device and supports a maximum of 24 interfaces each of which has its own IP address.
- **Access point:** Access point (AP) is a layer-2 wireless device working per 802.11 Wi-Fi protocol. It can be connected to wireless nodes via wireless links and to a router or a switch via a wired link.

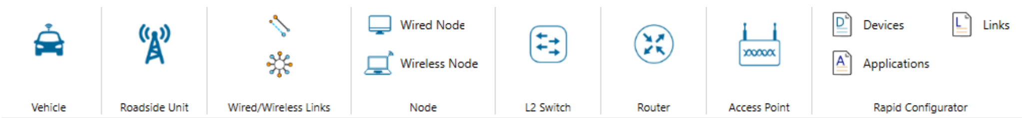


Figure 2-3: *VANET Library Device Palette in GUI*

2.3 Set Node, Link and Application Properties

- Right click on the appropriate node or link and select Properties. Then modify the parameters according to the requirements.
- Routing Protocol in Network Layer and all user editable properties in Data Link Layer few properties are Global or Local, Physical Layer and Power are Local.
- Physical layer standards are acting as Link global.
- In the General properties, Mobility model is set to SUMO, and it is editable. This signifies that the Node movements will be traced from SUMO.
- File name gives the path to Sumo Configuration file that was given by the user.
- Step Size is taken from the Sumo Configuration file specified which tells the amount of time paused in sumo corresponding to a single step of SUMO Simulation.
- In Interface (wireless) properties, under Physical layer, by default Standard is set to IEEE 802.11p in case of VANET.
- The following are the important properties of VANET Wireless Node (RSU/Vehicle) in Data link and Physical layers.

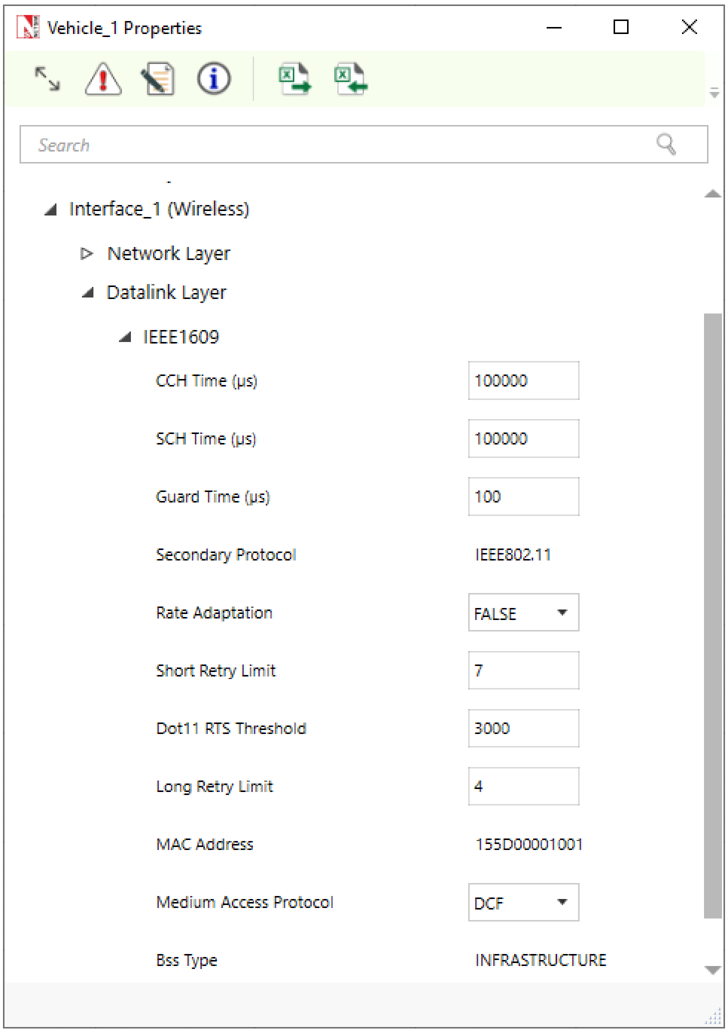


Figure 2-4: *Vanet > Datalink layer properties window*

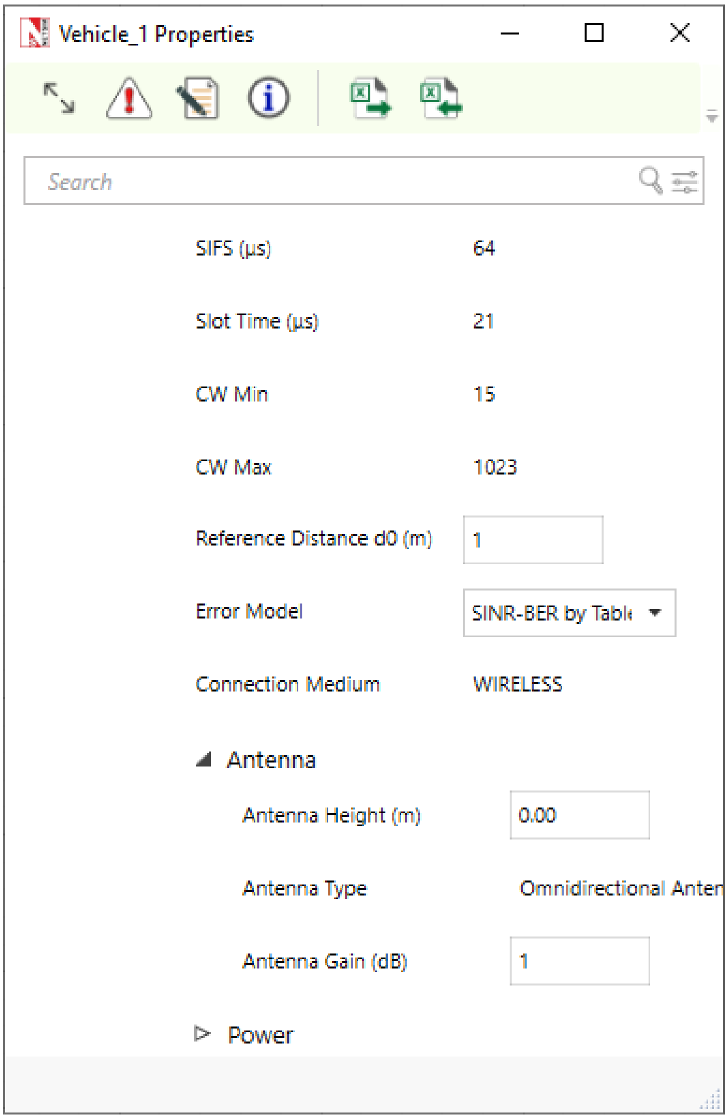


Figure 2-5: *Vanet > Physical layer properties window*

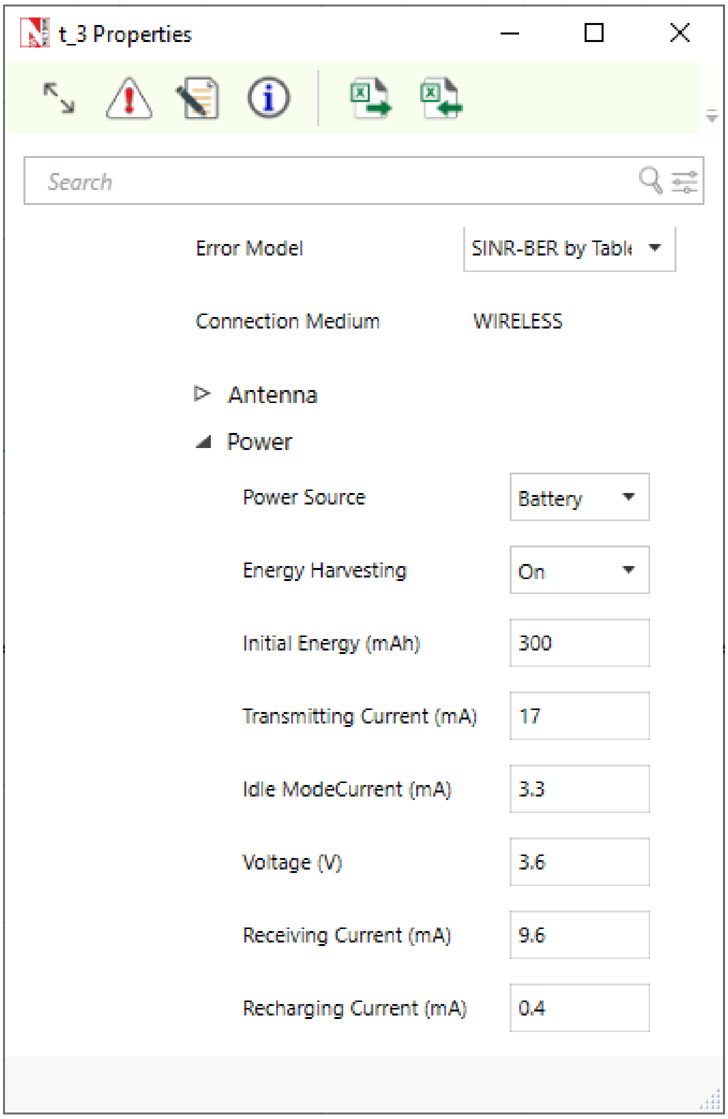


Figure 2-6: *Vanet > Physical layer properties window > Battery model*

- Click on the Application icon present in the Set Traffic option and set properties. Multiple applications can be generated by using add button in Application properties.

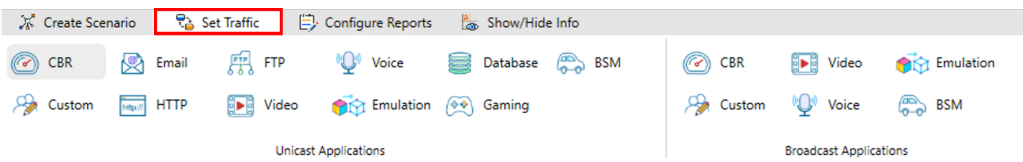


Figure 2-7: *Application icon present on top ribbon*

- Set the values according to requirement and click OK.

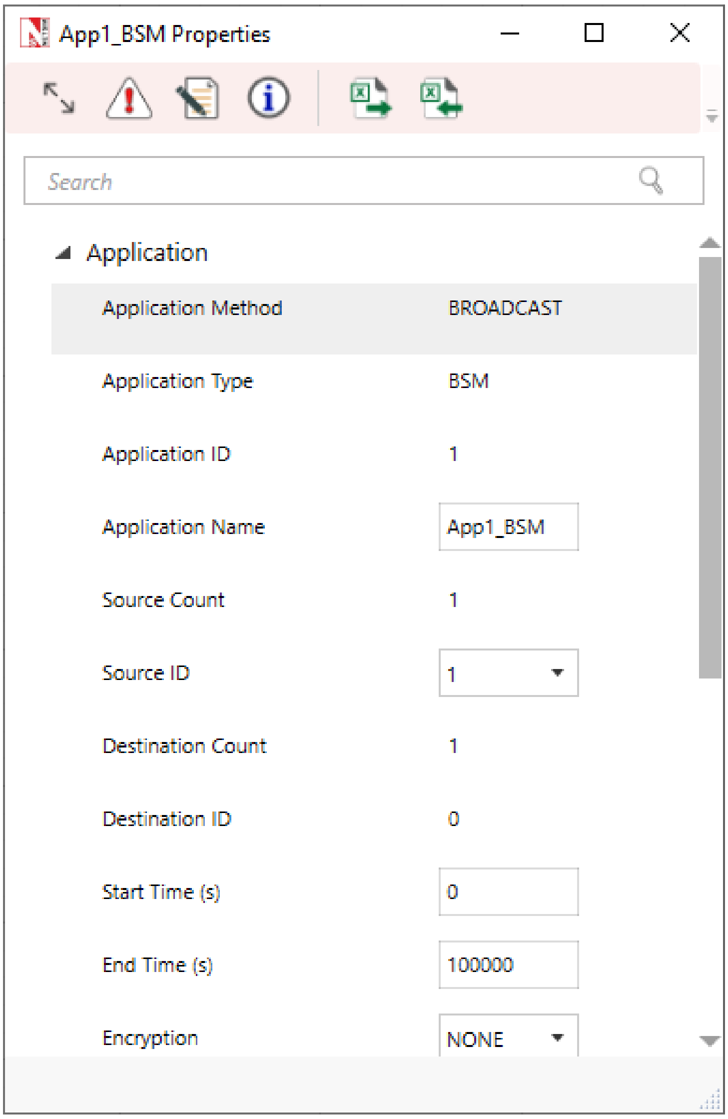


Figure 2-8: *Application Configuration window*

Detailed information on Application properties is available in section 6 of NetSim User Manual.

2.4 Enable Packet Trace, Event Trace & Plots (Optional)

For detailed packet information, enable packet and event tracing before running the simulation. To enable trace files, click on Configure Reports in the top ribbon, check the Packet Trace and Event Trace checkboxes. For further analysis, please refer to sections 8.4 and 8.5 in User Manual.

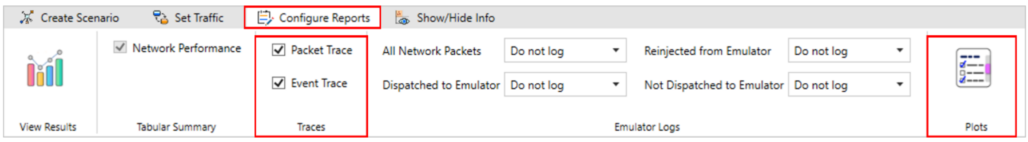


Figure 2-9: *Enable Packet Trace, Event Trace & Plots options on top ribbon.*

2.5 Enable protocol specific logs and plots

NetSim provides protocol-specific logs for VANET libraries, which users can enable before running a simulation. These can be enabled by clicking on configure reports in top ribbon ¿ clicking on plots ¿ choosing as desired and running the simulation.

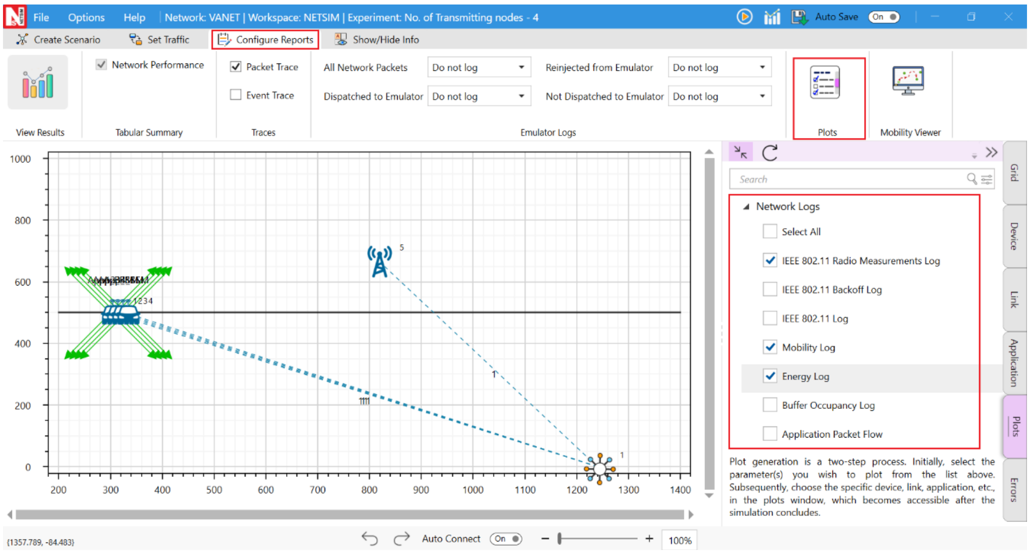


Figure 2-10: *Enabling IEEE 802.11 Radio measurements log in VANETs*

Similarly, users can enable the plots for Wi-Fi radio measurements and energy.

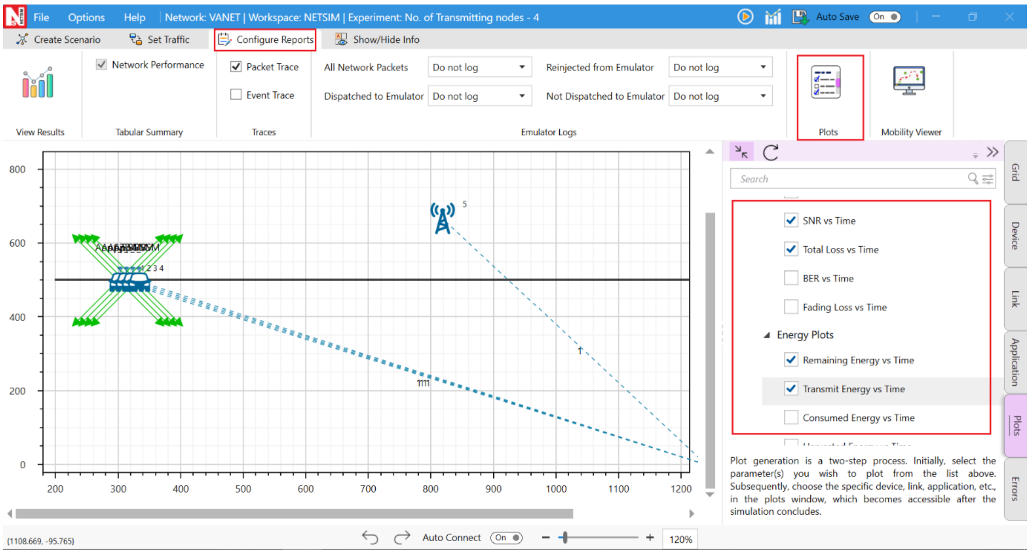


Figure 2-11: *Enabling plots for VANETs*

2.6 GUI Configuration Parameters

Table 2-1: *Datalink layer, Network layer and Physical layer properties for Vehicles and RSUs*

Parameter	Scope	Range	Description
Vehicle ¿ Interface(Wireless) – Datalink Layer			
CCH Time (μs)	Global	0–1000000	A radio channel, intended for the exchange of management information. In NetSim when a BSM (safety) application is configured, it is transmitted on the CCH.
SCH Time (μs)	Global	0–1000000	A radio channel used for non-safety applications. In NetSim, when non safety applications such as CBR, Voice, Video, FTP etc., are configured, they are transmitted on the SCH.
Guard Interval (μs)	Global	1–100	A time interval at the start of each control channel (CCH) interval and service channel (SCH) interval during which devices that are switching channels do not transmit.
Rate Adaptation	Cell	False	The algorithm is similar to Receiver based Auto Rate (RBAR) algorithm. In this, the PHY rate gets set based on the target PEP (packet error probability) for a given packet size. The adaptation is termed as "FALSE" since the rate is pre-determined as per standard and there is no subsequent "adaptation".
Rate Adaptation		Minstrel	Rate adaptation algorithm implemented in Linux.
Rate Adaptation		Generic	The algorithm is similar to the Auto Rate FallBack (ARF) algorithm. In this algorithm (i) Rate goes up one step for 20 consecutive packet successes, and (ii) Rate goes down one step after 3 consecutive packet failures.
Short Retry Limit	Local	1 to 255	Determines the maximum number of transmission attempts of a frame. The length of MPDU is less than/equal to Dot11 RTS Threshold value, made before a failure condition is indicated.
Long Retry Limit	Local	1 to 255	Determines the maximum number of transmission attempts of a frame. The length of MPDU is greater than Dot11 RTS Threshold value, made before a failure condition is indicated.
Dot11 RTS Threshold	Local	0 to 4692480	The size of packets (or A-MPDU if applicable) above which RTS/CTS (Request to Send / Clear to Send) mechanism gets triggered.
MAC Address	Fixed	Auto Generated	The MAC address is a unique value associated with a network adapter. This is also known as hardware address or physical address. This is a 12-digit hexadecimal number (48 bits in length).

Continued on next page

Parameter	Scope	Range	Description
Physical Type	Global	DSSS	Direct Sequence Spread Spectrum. The physical type of parameter is set to DSSS if the standard selected is IEEE802.11b.
Physical Type	Global	OFDM	Orthogonal Frequency Division Multiplexing is utilized as a digital multicarrier modulation method. The physical type of parameter is set to OFDM if the standard selected is IEEE802.11a, g and p.
Physical Type	Global	HT	Operates in frequency bands 2.4GHz or 5GHz band. The physical type parameter is set to HT if the standard selected is IEEE802.11n.
Physical Type	Global	VHT	The physical type parameter is set to VHT if the standard selected is IEEE802.11ac.
Medium Access Protocol	Local	DCF	DCF is the process by which CSMA/CA is applied to Wi-Fi networks. DCF defines four components to ensure devices share the medium equally: Physical Carrier Sense, Virtual Carrier Sense, Random Back-off timers, and Interframe Spaces (IFS). DCF is used in non-QoS WLANs.
Medium Access Protocol	Local	EDCAF	QoS was introduced in 802.11e and is achieved using enhanced distributed channel access functions (EDCAFs). EDCA provides differentiated priorities to transmitted traffic, using four different access categories (ACs). With EDCA, high-priority traffic has a higher chance of being sent than low-priority traffic: a station with high priority traffic waits a little less before it sends its packet, on average, than a station with low priority traffic.
OCBA Activated	Local	True or False	This parameter determines the type of standard to be chosen for the OFDM physical type. The standard is set to IEEE802.11p if OCBA is True. The standard is set to IEEE802.11a and g if OCBA is False.
BSS Type	Fixed	Auto Generated	The BSS type is fixed to Infrastructure mode. The wireless device can communicate with each other or with a wired network.
Interface(Wireless) – Physical Layer			
Protocol	Fixed	IEEE 802.11	Defines the MAC and PHY specifications like IEEE802.11a/b/g/n/ac/p for wireless connectivity for fixed, portable and moving stations within a local area.

Continued on next page

Parameter	Scope	Range	Description
Connection Medium	Fixed	Auto Generated	Defines how the devices are connected or linked to each other.
Standard	Cell	IEEE 802.11 a/b/g/n/ac/p	Refers to a family of specifications developed by IEEE for WLAN technology. The IEEE standards supported in NetSim are IEEE 802.11 a, b, g, n, ac and p. 802.11a provides up to 54 Mbps in the 5GHz band. 802.11b provides 11 Mbps in the 2.4GHz bands. 802.11g provides 54 Mbps transmission over short distances in the 2.4 GHz band. 802.11n adds up MIMO. 802.11ac provides support for wider channels and beamforming capabilities. 802.11p provides support to Intelligent Transportation Systems.
Transmission Type	Fixed	DSSS	The transmission type parameter is DSSS if the standard selected is IEEE802.11b.
Transmission Type	Fixed	OFDM	The transmission type parameter is OFDM if the standard selected is IEEE802.11a, g and p.
Transmission Type	Fixed	HT	The transmission type parameter is HT if the standard selected is IEEE802.11n.
Transmission Type	Fixed	VHT	The transmission type parameter is VHT if the standard selected is IEEE802.11ac.
Transmit Power	Local	0 to 1000	Transmitted signal power. Note that the transmit power is not split among the antennas. This value is applied to each antenna in a multi-antenna transmitter. Unit is mW.
CCA Mode	Fixed	Auto Generated	A mechanism to determine whether a medium is idle or not. It includes Carrier sensing and energy detection.
Frequency Band	Cell	2.4, 5, 5.9 (Depends on the standard chosen)	Range of frequencies at which the device operates. The frequency band depends on the standard selected. Unit is GHz.
Bandwidth	Cell	5, 10, 20, 40, 60, 80, 160 (Depends on the standard chosen)	The bandwidth depends on the standard and the frequency band selected. Unit is MHz.
Standard Channel	Local	Depends on the standard chosen	The channel options defined in the standards. The options would also depend on the frequency band if the standard supports multiple bands.

Continued on next page

Parameter	Scope	Range	Description
Standard Channel SCH	Local	–	SCHs are dedicated communication channels used for transmitting data and information between vehicles and RSUs in a VANET. The Standard Service Channels (SCHs) in VANETs are primarily used for general data exchange, including applications such as traffic information sharing, multimedia data transmission, and other non-safety-critical communications. Supported Channels: These are the specific channels designated for data transmission in the VANET. The list includes channels like SCH 172 (5860 MHz), SCH 174 (5870 MHz), SCH 176 (5880 MHz), 180 (5900 MHz), 182 (5910 MHz) and 184 (5920 MHz).
Standard Channel CCH	Local	CCH 178 (5890 MHz)	The Control Channel (CCH) serves as a specialized communication channel dedicated to the management and control of VANET operations. CCH is primarily used for disseminating safety-related information and network management messages. It plays a crucial role in supporting applications like collision avoidance and traffic management. The Control Channel in VANET operates at channel CCH 178 (5890 MHz).
SIFS	Fixed	Auto Generated	The time interval required by a wireless device in between receiving a frame and responding to the frame. Unit is microseconds.
Slot Time	Fixed	Auto Generated	Time is quantized as slots in Wi-Fi. Unit is microseconds.
Guard Interval	Local	400 and 800	Guard Interval is intended to avoid signal loss from multipath effect. Unit is nanoseconds.
MCS Selection	Local	Auto Rate Fallback, Fixed	MCS selection in Wi-Fi impacts data rates and efficiency. Auto Rate Fallback adapts the MCS based on signal quality. Fixed MCS locks the MCS. Default Value: Auto Rate.
Data MCS	Local	802.11b: 0–3, 802.11a/g/p: 0–7, 802.11n: 0–7, 802.11ac: 0–9 (MCS 9 not available for 20MHz in VHT)	Allows selection of the MCS value for different Wi-Fi standards. Determines the modulation and coding scheme. Default Value: 0.

Continued on next page

Parameter	Scope	Range	Description
Data PHY Rate (Mbps)	Local	Determined by selected Data MCS and Wi-Fi standard	Shows the physical layer data rate based on the chosen modulation and coding scheme. (MCS)
CW Min	Fixed	Auto Generated	The minimum size of the Contention Window in units of slot time. The CW min is used by the MAC to calculate the back off time for channel access during a carrier sense.
CW Max	Fixed	Auto Generated	The maximum size of the Contention Window in units of slot time. The CW is doubled progressively when collisions occur.
Error Model	Local	SINR-BER-By-Table, SINR-BER-By-Formula	Specifies how the Bit Error Rate (BER) is calculated: BER is determined based on predefined tables mapping SINR to BER. BER is calculated using mathematical formulas that account for the modulation and coding schemes used, based on the SINR value.
Antenna Height	Local	0 to 100m	It is used in the pathloss calculation in the following models: Cost231 Hata Urban, Cost231 Hata SubUrban, Hata Urban, Hata SubUrban and Two Ray. This parameter has no effect when using any of the other pathloss models. Default: 0.0 m.
Antenna Gain	Local	0 to 1000 dB	A relative measure of an antenna's ability to direct or concentrate radio frequency energy in a particular direction or pattern. The measurement is typically measured in dBi (Decibels relative to an isotropic radiator).
Antenna Type	Fixed	–	NetSim VANET supports one type of Antenna, Omnidirectional Antenna.
Power Model			
Power Source	Local	Main Line or Battery	VANETs communicate with each other using battery power. By default, the power model is set to Main Line, which represents a general-purpose alternating current (AC) electric power supply. The power model is user-configurable, with adjustable properties.

Continued on next page

Parameter	Scope	Range	Description
Energy Harvesting	Local	On or Off	Energy harvesting is the process of deriving energy from external sources (e.g., solar power, thermal energy, wind energy, and kinetic energy), capturing it, and storing it for use in small, wireless autonomous devices, such as those in wearable electronics and wireless sensor networks. NetSim supports an abstract energy harvesting model in which a specified amount of energy (calculated from the recharging current and specified voltage) is periodically added to the remaining energy of the node to replenish the battery. This feature can be turned on or off.
Initial Energy	Local	0.001–1000 mAh	A node has an initial value which is the level of energy the node has at the beginning of the simulation.
Transmitting Current	Local	0–5000 mA	In the Transmitting mode (Tx mode), the node consumes energy to transfer packets or data. The amount of energy consumed in this mode depends on the number of packets sent by the node, greater the number of packets, the more energy is consumed.
Idle Mode Current	Local	0–500 mA	In idle mode, a node doesn't transmit or receive data but still listens to the wireless medium for potential packets and new nodes. This consumes less energy than sending or receiving, as no active communication occurs.
Voltage	Local	0–10 V	Voltage is a measure of the energy carried by the charge.
Receiving Current	Local	0–1000 mA	In the Receiving mode (Rx mode), the nodes are actively listening to the incoming data, it consumes the energy as it receives the data from the sender.
Recharging Current	Local	0–20 mA	Recharging Current refers to the flow of electric charge supplied to a battery during the recharging process.
Network Layer			
Routing Protocol	Global	DSR, AODV, ZRP, OLSR	DSR, AODV, ZRP and OLSR routing protocols are supported.
DSR Routing Protocol			

Continued on next page

Parameter	Scope	Range	Description
ACK Type	Global	LINK_LAYER_ACK or NETWORK_LAYER_ACK	The user can enable either Link Layer ACK (Layer 2 ACK) or Network Layer ACK (Layer 3 ACK). Link Layer ACK uses MAC layer acknowledgment for route maintenance, while Network Layer ACK uses DSR acknowledgment for route maintenance. For more details, refer to sections 3.2.1 and 3.2.2 of the MANET Technology Library.
AODV Routing Protocol			
Hello Message	Global	Enable/Disable	Hello messages are periodic broadcasts used to maintain local connectivity and discover neighbors, ensuring that nodes are aware of each other's presence. Enabled: You will observe Hello packets being sent and received in the simulation. Disabled: Hello packets are not transmitted or received. Without HELLO messages, AODV's route discovery (RREQ/RREP) remains the same. However, route maintenance shifts from proactive local link sensing (via HELLO) to reactive link break detection.
ZRP and OLSR Routing Protocol			
Hello Interval	Global	1–100 s	Hello interval parameter is used for neighbor discovery process. This parameter determines how frequently Hello messages are sent out and also how frequently a neighbor table will be updated.
Refresh Interval	Global	1–100 s	Refresh interval is the duration after which each active node periodically refreshes routes to itself.
IARP	Fixed	–	IARP is used by a node to communicate with the interior nodes of its zone and is limited by the zone radius.
TC Interval	Global	1–100 s	Topology Control messages are the link state signaling done by OLSR. These messages are sent at TC interval every time.
Zone radius	Global	2–225 m	Zone radius parameter is present for ZRP Protocol. ZRP divides the entire network into zones. The radius of these zones is defined by Zone radius.

2.7 Run Simulation

Click on Run Simulation icon on the top toolbar. Simulation time is set from the Configuration file of Sumo. The simulation has three options.

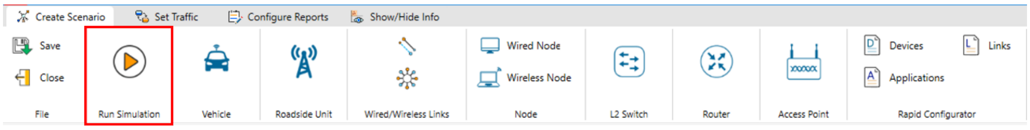


Figure 2-12: *Run Simulation option on top ribbon*

SUMO determines the positions of vehicles with respect to time as per the road conditions. NetSim reads the coordinates of vehicles from SUMO (through pipe) during runtime and uses it as input for vehicle mobility.

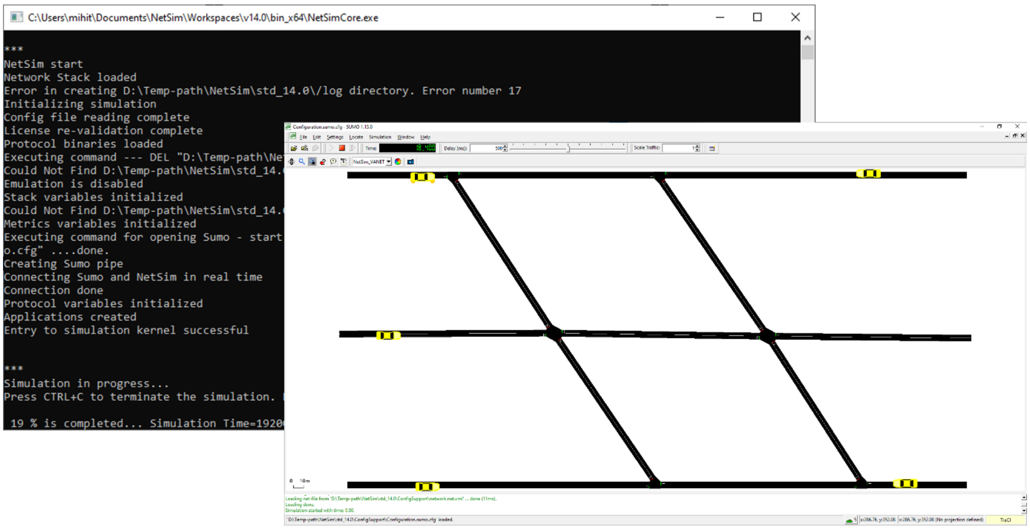


Figure 2-13: *NetSim simulation window and SUMO simulation window runs simultaneously*

Users can see the movement of vehicles and observe vehicular simulation in SUMO and observe equivalent simulation in NetSim.

3 Model Features

3.1 Implementation of the 802.11p in NetSim

- The Ad hoc Wi-fi MAC allows for STA transmissions of data frames outside-the-context-of-a-BSS (OCB). Establishing a secure BSS necessitates announcement, scanning, synchronization, and association and the time required is extremely undesirable in vehicular environments. NetSim therefore allows for direct and instantaneous link setups with no establishment of a BSS. There is no authentication nor association.
- Supports a channel bandwidth of 5MHz, 10 MHz, 20MHz in the 5.9 GHz band
- Supported PHY rates are as follows:
  - 1.5, 2.25, 3, 4.5, 6, 9, 12 and 13.5 Mbps for 5MHz
  - 3, 4.5, 6, 9, 12, 18, 24 and 27 Mbps for 10MHz.
  - 6, 9, 12, 18, 24, 36, 48 and 54 Mbps for 20MHz.

- The PHY rates for IEEE 802.11p in NetSim are determined from the MCS tables of IEEE 802.11-OFDM (provided below). The receiver measures the received power and is assumed to transmit this information to the transmitter instantaneously and out of band. The transmitter compares the received power to the receiver sensitivity and selects the MCS where the received power meets or exceeds the sensitivity.

Table 3-1: *MCS table for 5 MHz bandwidth*

MCS Index	Receiver Sensitivity (dBm)	Data Rate (Mbps)	Modulation Technique	Coding Rate
1	−88	1.5	BPSK	1/2
2	−87	2.25	BPSK	3/4
3	−85	3	QPSK	1/2
4	−83	4.5	QPSK	3/4
5	−80	6	16_QAM	1/2
6	−76	9	16_QAM	3/4
7	−72	12	64_QAM	2/3
8	−71	13.5	64_QAM	3/4

Table 3-2: *MCS table for 10 MHz bandwidth*

MCS Index	Receiver Sensitivity (dBm)	Data Rate (Mbps)	Modulation Technique	Coding Rate
1	−85	3	BPSK	1/2
2	−84	4.5	BPSK	3/4
3	−82	6	QPSK	1/2
4	−80	9	QPSK	3/4
5	−77	12	16_QAM	1/2
6	−73	18	16_QAM	3/4
7	−69	24	64_QAM	2/3
8	−68	27	64_QAM	3/4

Table 3-3: *MCS table for 20 MHz bandwidth*

MCS Index	Receiver Sensitivity (dBm)	Data Rate (Mbps)	Modulation Technique	Coding Rate
1	−82	6	BPSK	1/2
2	−81	9	BPSK	3/4
3	−79	12	QPSK	1/2
4	−77	18	QPSK	3/4
5	−74	24	16_QAM	1/2
6	−70	36	16_QAM	3/4
7	−66	48	64_QAM	2/3
8	−65	54	64_QAM	3/4

- Transmission type is OFDM
  - with slot time equal to 9 μs and SIFS equal to 16 μs for 20 MHz
  - with slot time equal to 13 μs and SIFS equal to 32 μs for 10 MHz
  - with slot time equal to 21 μs and SIFS equal to 64 μs for 5 MHz
- Uses a Medium Access Control (MAC) protocol based on the Carrier Sense Multiple Access Collision Avoidance (CSMA/CA) protocol, which is explained below.

1. When a node wants to send a message, the channel must be idle for a duration of SIFS. If the channel is idle, it starts transmission.

2. When a node finds the channel busy, it chooses a random backoff time from the interval $[0, CW]$ and transmits only when the backoff timer has elapsed. The variable CW represents the size of the Contention Window.
3. When the SCH is used and a node does not receive an acknowledgement for a message, it concludes that the message has collided and is lost, so the value of CW is doubled, and it will retry transmission.
4. In the CCH however, beacons are broadcast in the channel and no acknowledgments are sent. Therefore, the value of CW is never doubled in the CCH.

3.2 Introduction of IEEE 802.11bd

IEEE 802.11bd is a vehicular communication standard for Intelligent Transportation Systems (ITS). It is defined as an amendment to IEEE 802.11, operating in the 5.9 GHz band. It supports coexistence with IEEE 802.11p (non-NGV) stations and introduces enhancements to the PHY layer for next generation V2X communication.

NetSim's VANETs module supports IEEE 802.11bd. The following enhancements are defined in IEEE 802.11bd over IEEE 802.11p:

- Spatial stream support: up to 2 transmitting and 2 receiving antennas, supporting spatial diversity.
- Frame Aggregation: multiple MPDUs combined into a single PHY transmission, improving channel utilization.
- Guard Interval fixed at 1600 ns, as defined in IEEE 802.11bd (Table 32-6).
- Bandwidth support for both 10 MHz and 20 MHz channels.
- Extended MCS range: NGV-MCS 0–8 (up to 256-QAM), with MCS 9 not valid at 10 MHz.

All other PHY and MAC properties follow the IEEE 802.11bd specification.

3.3 Implementation of IEEE 802.11bd in NetSim

The IEEE 802.11bd implementation in NetSim includes vehicular-specific extensions to the standard PHY and MAC. Key aspects are described below.

3.3.1 PHY Layer

- Operates in the 5.9 GHz band with 10 MHz and 20 MHz channel bandwidths.
- Transmission type: BD.
- Slot time (aSlotTime): 13 μs (Table 32-20, IEEE 802.11bd-2022).
- SIFS (aSIFSTime): 32 μs (Table 32-20, IEEE 802.11bd-2022).
- Guard Interval (T_{GI}): 1600 ns = $T_{DFT}/4$, where $T_{DFT} = 6.4$ μs (Table 32-6, IEEE 802.11bd-2022).
- Up to 2 transmitting and 2 receiving antennas (NSS up to 2).
- Frame Aggregation: 1 to 32 frames per PHY transmission.

PHY data rates are derived from the NGV-MCS tables defined in IEEE 802.11bd-2022. The receiver selects the MCS at which received power meets or exceeds the receiver sensitivity threshold (Table 32-16, IEEE 802.11bd-2022).

NGV-MCS Table — 10 MHz, NSS = 1 Source: struPhyParameters_10MHz_1NSS, IEEE 802.11bd-2022 Tables 32-21 and 32-16.

Table 3-4: *NGV-MCS Parameters — 10 MHz, NSS = 1*

NGV-MCS Index	Modulation	Coding Rate (R)	Data Rate (Mbps) 1.6 μs GI	Rx Sensitivity (dBm)
0	BPSK	1/2	3.3	−85
1	QPSK	1/2	6.5	−82
2	QPSK	3/4	9.8	−80
3	16-QAM	1/2	13.0	−77
4	16-QAM	3/4	19.5	−73
5	64-QAM	2/3	26.0	−69
6	64-QAM	3/4	29.3	−68
7	64-QAM	5/6	32.5	−67
8	256-QAM	3/4	39.0	−62
9	256-QAM	5/6	(Not Valid)	−57

NGV-MCS Table — 10 MHz, NSS = 2 Source: struPhyParameters_10MHz_2NSS, IEEE 802.11bd-2022 Tables 32-22 and 32-16.

Table 3-5: *NGV-MCS Parameters — 10 MHz, NSS = 2*

NGV-MCS Index	Modulation	Coding Rate (R)	Data Rate (Mbps) 1.6 μs GI	Rx Sensitivity (dBm)
0	BPSK	1/2	6.5	−85
1	QPSK	1/2	13.0	−82
2	QPSK	3/4	19.5	−80
3	16-QAM	1/2	26.0	−77
4	16-QAM	3/4	39.0	−73
5	64-QAM	2/3	52.0	−69
6	64-QAM	3/4	58.5	−68
7	64-QAM	5/6	65.0	−67
8	256-QAM	3/4	78.0	−62
9	256-QAM	5/6	(Not Valid)	−57

NGV-MCS Table — 20 MHz, NSS = 1 Source: struPhyParameters_20MHz_1NSS, IEEE 802.11bd-2022 Tables 32-23 and 32-16.

Table 3-6: *NGV-MCS Parameters — 20 MHz, NSS = 1*

NGV-MCS Index	Modulation	Coding Rate (R)	Data Rate (Mbps) 1.6 μs GI	Rx Sensitivity (dBm)
0	BPSK	1/2	6.8	−82
1	QPSK	1/2	13.5	−79
2	QPSK	3/4	20.3	−77
3	16-QAM	1/2	27.0	−74
4	16-QAM	3/4	40.5	−70
5	64-QAM	2/3	54.0	−66
6	64-QAM	3/4	60.8	−65
7	64-QAM	5/6	67.5	−64
8	256-QAM	3/4	81.0	−59
9	256-QAM	5/6	90.0	−57

NGV-MCS Table — 20 MHz, NSS = 2 Source: struPhyParameters_20MHz_2NSS, IEEE 802.11bd-2022 Tables 32-24 and 32-16.

Table 3-7: *NGV-MCS Parameters — 20 MHz, NSS = 2*

NGV-MCS Index	Modulation	Coding Rate (R)	Data Rate (Mbps) 1.6 μs GI	Rx Sensitivity (dBm)
0	BPSK	1/2	13.5	−82
1	QPSK	1/2	27.0	−79
2	QPSK	3/4	40.5	−77
3	16-QAM	1/2	54.0	−74
4	16-QAM	3/4	81.0	−70
5	64-QAM	2/3	106.0	−66
6	64-QAM	3/4	121.5	−65
7	64-QAM	5/6	135.0	−64
8	256-QAM	3/4	162.0	−59
9	256-QAM	5/6	180.0	−57

3.4 GUI Configuration Parameters

The following tables list the configurable parameters for IEEE 802.11bd Vehicles and RSUs in NetSim, organized by layer.

3.4.1 Physical Layer Parameters

Table 3-8: *Physical Layer Properties — IEEE 802.11bd Vehicle/RSU*

Parameter	Scope	Range	Description
Standard	Cell	IEEE 802.11bd	IEEE 802.11bd is a vehicular communication standard for Intelligent Transportation Systems (ITS). It operates in the 5.9 GHz band and is backward-compatible with IEEE 802.11p. Enhancements include multi-antenna support, frame aggregation, and extended SCH channel range.
Transmit Power	Local	0–1000 mW	Transmitted signal power per antenna. The transmit power is not split among antennas; the configured value is applied independently to each antenna in a multi-antenna transmitter. Unit: mW. Default: 1000 mW.
No. of Frames to Aggregate	Local	1–32	Number of MAC Protocol Data Units (MPDUs) combined into a single PHY-layer transmission. A value of 1 disables aggregation. Default: 1.
Frequency Band	Fixed	5.9 GHz	IEEE 802.11bd operates in the 5.9 GHz band, as specified by DSRC/WAVE standards. This parameter is not configurable.
Bandwidth	Cell	10 MHz, 20 MHz	Channel bandwidth. Determines available data rates and spectral occupancy. Default: 10 MHz.
SCH Channel	Local	172_5860 – 184_5920	Service Channel (SCH) used for non-safety data applications. The SCH channel plan is defined by IEEE 1609.4 and is the same as IEEE 802.11p. IEEE 802.11bd does not define a new channel plan.

Continued on next page

Parameter	Scope	Range	Description
CCH Channel	Fixed	CCH 178 (5890 MHz)	Control Channel (CCH) used for safety management messages, including Basic Safety Messages (BSM) and WAVE Service Advertisements (WSA). Fixed at 5890 MHz (Channel 178) per IEEE 1609.4. Not configurable.
Guard Interval	Fixed	1600 ns	Guard interval between OFDM symbols. IEEE 802.11bd defines $T_{GI} = 1.6$ μs $(= T_{DFT}/4)$, where $T_{DFT} = 6.4$ μs. Protects against inter-symbol interference from multipath propagation. Not configurable.
Connection Medium	Fixed	Wireless	IEEE 802.11bd nodes communicate over the wireless medium in OCB (Outside the Context of a BSS) mode, without BSS establishment, authentication, or association.
Error Model	Local	SINR-BER-By-Table, SINR-BER-By-Formula	Specifies the method used to compute Bit Error Rate (BER) during simulation. SINR-BER-By-Table: BER is determined from pre-computed lookup tables mapping SINR to BER per MCS. SINR-BER-By-Formula: BER is computed from mathematical expressions based on SINR and MCS.

3.4.2 Antenna Parameters

Table 3-9: *Antenna Properties — IEEE 802.11bd Vehicle/RSU*

Parameter	Scope	Range	Description
Antenna Height	Local	0–100 m	Used in path loss calculations for the following models: Cost231 Hata Urban, Cost231 Hata SubUrban, Hata Urban, Hata SubUrban, and Two Ray. Has no effect when using other path loss models. Default: 0.0 m.
Antenna Type	Fixed	Omnidirectional	IEEE 802.11bd uses an omnidirectional antenna, which radiates and receives signals uniformly in the horizontal plane.
Antenna Gain	Local	0–1000 dB	Measure of the antenna's ability to direct radio frequency energy relative to an isotropic radiator. Expressed in dBi. Default: 0 dB.
Transmitting Antennas	Local	1 or 2	Number of transmitting antennas. IEEE 802.11bd supports up to 2, enabling spatial multiplexing or transmit diversity. IEEE 802.11p supports only 1.
Receiving Antennas	Local	1 or 2	Number of receiving antennas. IEEE 802.11bd supports up to 2, enabling receive diversity and MIMO reception.

3.4.3 Power Parameters

Table 3-10: *Power Properties — IEEE 802.11bd Vehicle/RSU*

Parameter	Scope	Range	Description
Power Source	Local	Main Line or Battery	Power source for the node. VANET nodes typically operate on battery. Default is Main Line (AC supply). Configurable with adjustable current and voltage properties.
Energy Harvesting	Local	On or Off	When enabled, the node periodically replenishes battery energy based on the configured recharging current and voltage. NetSim uses an abstract harvesting model.
Initial Energy	Local	0.001–1000 mAh	Battery charge at the start of the simulation.
Transmitting Current	Local	0–20 mA	Current drawn when the node is transmitting. Energy consumed scales with the number of packets sent.
Idle Mode Current	Local	0–20 mA	Current drawn when the node is idle (listening to the medium, not transmitting or receiving).
Receiving Current	Local	0–20 mA	Current drawn when the node is receiving data.
Recharging Current	Local	0–20 mA	Current supplied to the battery during energy harvesting.
Voltage	Local	0–10 V	Operating voltage of the node's power supply.

3.4.4 Frame Aggregation

Frame aggregation combines multiple MPDUs into a single PHY-layer transmission, improving channel utilization. The parameter is described in the table below.

Table 3-11: *Frame Aggregation Parameter — IEEE 802.11bd*

Parameter	Scope	Range	Description
No. of Frames to Aggregate	Local	1–32	Number of MPDUs combined into a single PHY-layer transmission. A value of 1 means no aggregation. Higher values improve throughput under good channel conditions. Default: 1.

3.5 MAC Layer

The MAC layer defined is same for both 802.11p and 802.11bd based on the standard IEEE 1609 protocol.

- Medium access uses CSMA/CA.
- DSRC channel switching between CCH and SCH is supported. CCH time, SCH time, and Guard Interval are user-configurable.
- BSM (Basic Safety Message) packets are transmitted on the CCH using WSMP over IEEE 1609.
- Non-safety application traffic (CBR, FTP, Voice, Video, etc.) is transmitted on the SCH.

- Frame aggregation is supported on the SCH.

3.6 DSRC Channels: CCH and SCH

Vehicles (OBUs) and RSUs can operate in (switch between) multiple channels i.e., in the SCH and CCH as explained below.

- **Control channel (CCH):** A radio channel, intended for the exchange of management information. In NetSim when a BSM (safety) application is configured, it is transmitted on the CCH.
- **Service channel (SCH):** These are radio channels used for non-safety application. In NetSim, when non safety application such as CBR, Voice, Video, FTP etc., are configured, they are transmitted on the SCH.
- **Guard interval:** A time interval at the start of each control channel (CCH) interval and service channel (SCH) interval during which devices cannot transmit.

Each synchronization interval SI is split as follows:

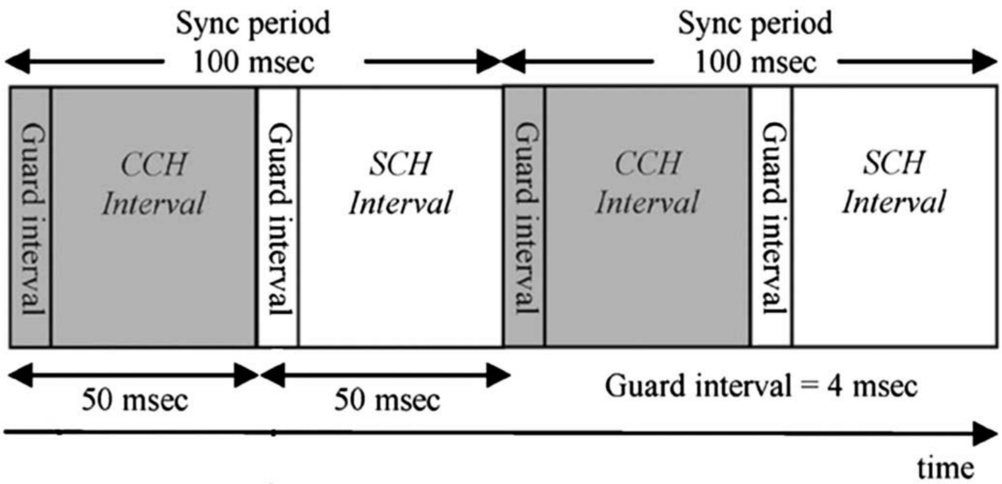


Figure 3-1: *We see the time divisions in DSRC. Each synchronization period consists of 1 CCH, 1 SCH and 1 guard interval. While the sync period is generally equal to 100 ms. NetSim allows users to modify the CCH and SCH interval, and in turn the total synchronization period.*

All devices (Vehicles and RSUs) switch between SCH and CCH and the alternation is based on the time divisions. NetSim allows the user to configure values of CCH interval, SCH interval and Guard interval. NetSim supports 6 service channels (SCH): 172 (5860 MHz), 174 (5870 MHz), 176 (5880 MHz), 180 (5900 MHz), 182 (5910 MHz) and 184 (5920 MHz), and 1 control channel (CCH): 178 (5890 MHz). The default channels used in NetSim are SCH 171 (5.855 GHz) and CCH 178 (5.890 GHz).

3.7 BSM Application

- DSRC protocol runs with BSM (Basic Safety Message) applications. BSM is a broadcast packet transmitted at a regular interval.
- The BSM Application class sends and receives the IEEE 1609 WAVE (Wireless Access in Vehicular Environments) Basic Safety Messages (BSMs). The BSM is a 20-byte packet that is generally broadcast from every vehicle at a nominal rate of 10 Hz. In NetSim this can be configured as a broadcast or as a unicast application. Note that a broadcast application can only be transmitted over a single hop. NetSim does not transmit broadcast applications over multiple hops.
- This application does not follow the IP stack. It runs WSMP protocol over IEEE 1609. There is no routing; static routes cannot be set, and packets are sent directly to the destination.

3.8 Differences from IEEE 802.11p

IEEE 802.11bd is backward-compatible with IEEE 802.11p. The table below lists the key differences between the two standards as implemented in NetSim.

Feature	IEEE 802.11p	IEEE 802.11bd
MAC/PHY Base	IEEE 802.11p	IEEE 802.11bd
Guard Interval (T_{GI})	800 ns	1600 ns (fixed)
Slot Time	13 μs	13 μs
SIFS	32 μs	32 μs
Transmitting Antennas	1	1 or 2
Receiving Antennas	1	1 or 2
Frame Aggregation	Not supported	1 to 32 frames
MCS Range	MCS 0–7 (up to 64-QAM)	NGV-MCS 0–8 (up to 256-QAM); MCS 9: valid at 20 MHz only
Peak Data Rate (10 MHz, 1 NSS)	27 Mbps	39 Mbps (NGV-MCS 8)
Peak Data Rate (20 MHz, 2 NSS)	54 Mbps	180 Mbps (NGV-MCS 9)
SCH Channels	channels (172–184)	channels (172–184) (same as 802.11p, per IEEE 1609.4)
CCH Channel	CCH 178 (5890 MHz)	CCH 178 (5890 MHz)
Bandwidth	5 MHz, 10 MHz, 20 MHz	10 MHz, 20 MHz
Frequency	5.9 GHz	5.9 GHz
DSRC/WAVE Protocol	Supported	Supported
BSM Application	Supported	Supported
SUMO Interfacing	Supported	Supported

3.9 NetSim – SUMO interfacing

- NetSim's VANET module allows users to interface with SUMO which is an open-source road traffic simulation package designed to handle vehicular & road networks. The road traffic simulation is done by SUMO while NetSim does the network simulation along with RF propagation modelling in the physical layer. While SUMO Simulates the road traffic conditions and movements, NetSim Simulates the communication occurring between the Vehicles.
- NetSim and SUMO are interfaced using 'pipes'. A pipe is a section of shared memory that processes use for communication. The SUMO process writes information to pipe, then NetSim process reads the information from pipe. On running the Simulation, SUMO determines the positions of vehicles with respect to time as per the road conditions. NetSim reads the coordinates of vehicles from SUMO (through pipes) in runtime and uses it as input for vehicle mobility.
- Users will notice an inversion along X axis in the NetSim GUI, since origin (0, 0) in SUMO is at the left bottom, while origin is at the left top in NetSim.
- VANET operates in a wireless environment and hence RF channel loss occurs. The amount of loss can be configured by users. To modify the Wireless channel characteristics users can right click on the ad hoc/wireless link and modify the channel characteristics as per the requirement.
- Source code related to interfacing of SUMO and NetSim is available in Sumo_interface.c file inside the mobility folder/project.

3.10 How to create a VANET using SUMO and simulate with NetSim

A SUMO network can be created either manually or using SUMO NetEdit.

3.10.1 Using SUMO NetEdit utility and randomtrips.py to configure road traffic models

Netedit is a road network editor for the road traffic simulation in SUMO. This utility enables users to efficiently design road networks and generate the corresponding network XML file, which is an essential component of the SUMO configuration.

Steps to create a simple SUMO network using netedit utility:

- **Step 1:** Open netedit from <SUMO_INSTALL_DIRECTORY>/bin (C:\Program Files (x86)\Eclipse\Sumo\bin) and select File → New Network

Refer SUMO Documentation: "https://sumo.dlr.de/docs/Netedit/index.html" for more details on modes of operation.

Figure 3-2: *SUMO NetEdit New Network Screen*

- **Step 2:** Select Creating junction and edges option as shown below or click on character "e" in the keyboard.

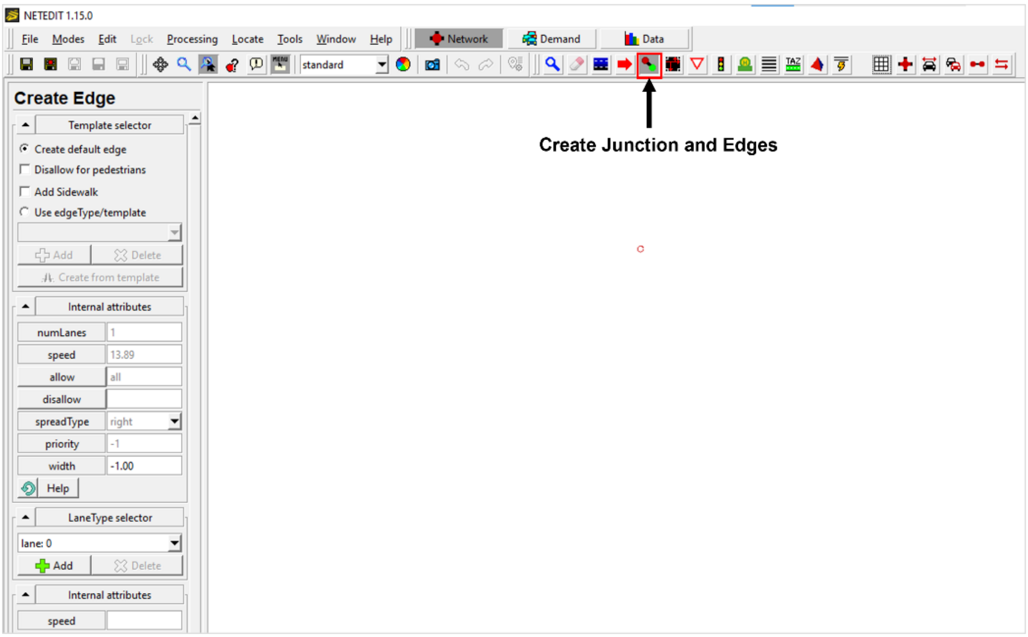


Figure 3-3: *SUMO NetEdit Screen*

- **Step 3:** Enable the check boxes "chain", "two-way" and "Grid" which are present in the right-side corner.

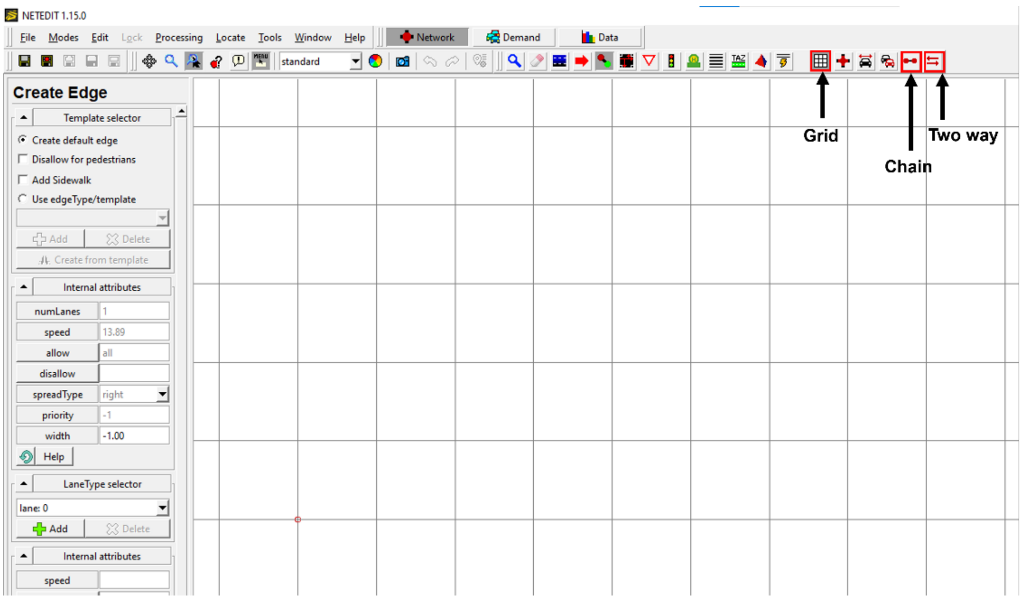


Figure 3-4: *SUMO NetEdit design window*

- **Step 4:** Adjust the design area to ensure that the road network lies in the Positive XY quadrant. This will help in avoiding complexities when opening the network scenario in NetSim.
- **Step 5:** Click on grid area to create edges, clicking again will create a new edge which will automatically get connected to the previous edge as shown below.

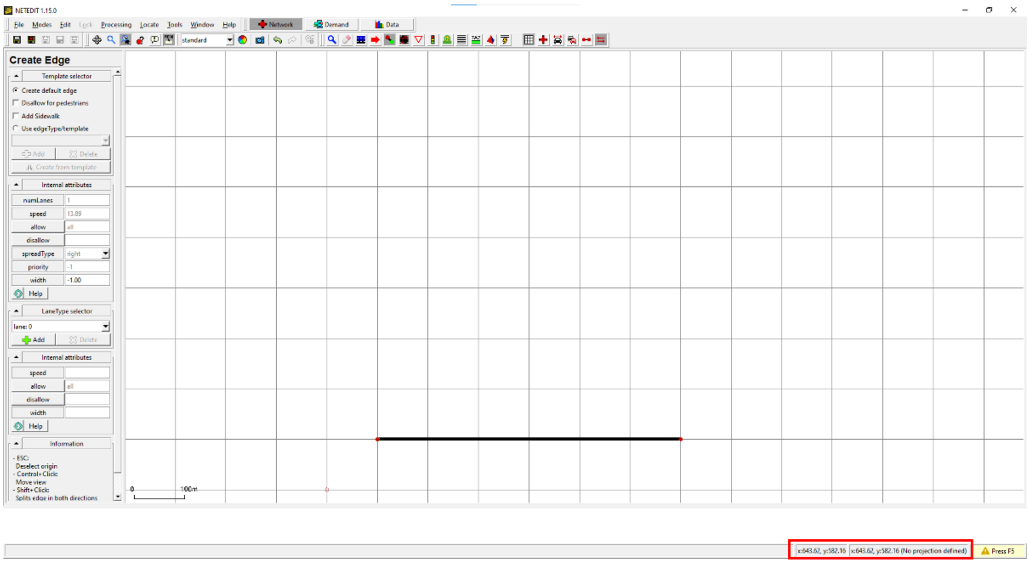


Figure 3-5: *Creating edges SUMO NetEdit design window*

- **Step 6:** Select “(t) Traffic Lights”. Select the junctions and click on Create TLS button on the left to add Traffic Signal to it.

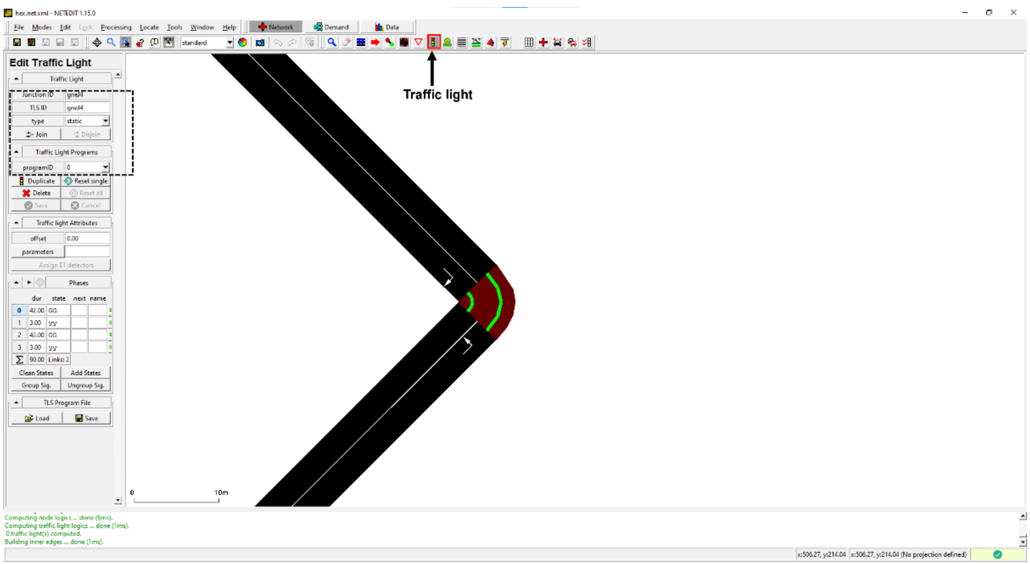


Figure 3-6: *Adding Traffic Signal to Network*

- **Step 7:** Select “(c) Connect” icon. Select the lanes and ensure connectivity between them.

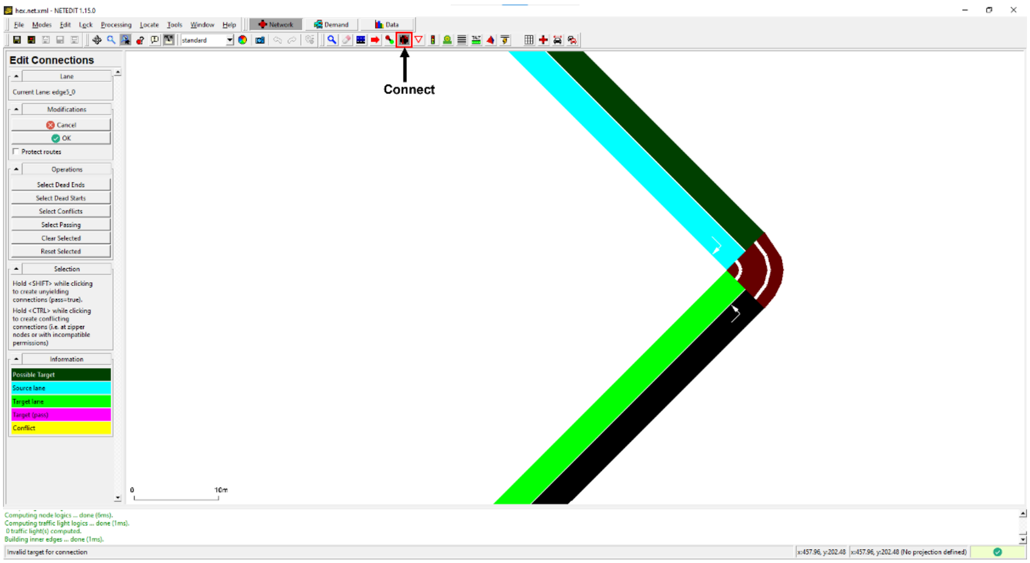


Figure 3-7: *Select the lanes and connectivity between them*

- **Step 8:** Create a new folder and save the network file (*.net.xml) over there, say with a name network.net.xml

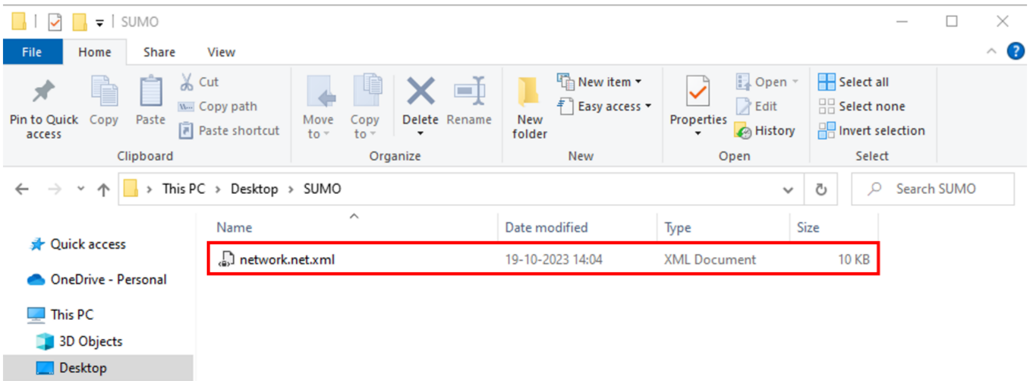


Figure 3-8: *network.net.xml inside the folder*

- **Step 9:** Open command prompt with the current working directory as the folder where you have saved the network file in the previous step.
- **Step 10:** Using randomtrips.py utility present in <SUMO_INSTALL_DIRECTORY>/tools directory create trips file with the command.

COMMAND SYNTAX:

```
>"C:\Program Files (x86)\Eclipse\Sumo\tools\randomTrips.py"
   -n "*.net.xml" -e <NO_OF_TRIPS> --route-file "trips.xml"
```

Example Command:

```
>"C:\Program Files (x86)\Eclipse\Sumo\tools\randomTrips.py"
   -n "network.net.xml" -e 5 --route-file "trips.xml"
```

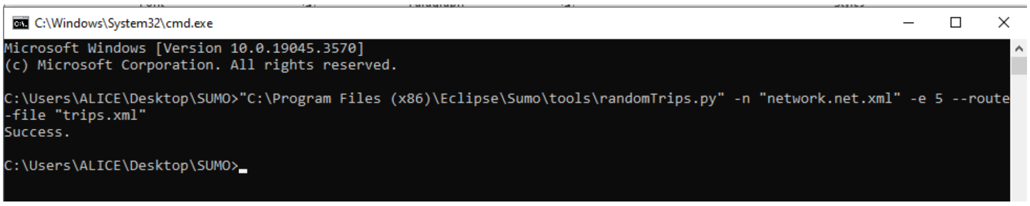


Figure 3-9: *Generating route file (trips.xml)*

This will create a trips file in your folder along with associated files.

- **Step 11:** Create a "file-settings.xml" file in your folder which contains the network and route file for improving the visualization of vehicles and their movements in a graphical user interface (GUI).

Following is a "file-settings.xml" file:

```
<viewsettings>
  <scheme name="NetSim_VANET">
    <vehicles vehicleQuality="2" vehicle_exaggeration="4.00">
    </vehicles>
  </scheme>
  <delay value="500"/>
</viewsettings>
```

- **Step 12:** Create a SUMO configuration file (*.sumo.cfg) which points to the network and trips file, in your folder which contains the network and route file.

Refer: http://sumo.dlr.de/wiki/Tutorials/Hello_Sumo

Include parameter (To Run in NetSim):

```
"<step-length value="0.4"/>"
```

Following is a sample SUMO Configuration:

```
<configuration>
    <input>
        <net-file value="network.net.xml"/>
        <route-files value="trips.xml"/>
        <gui-settings-file value="file-settings.xml"/>
    </input>
    <time>
        <begin value="0"/>
        <end value="100"/>
        <step-length value="0.4"/>
    </time>
</configuration>
```

NOTE: Save above content as Configuration.sumo.cfg

You can copy the above contents to create a SUMO configuration file in your folder.

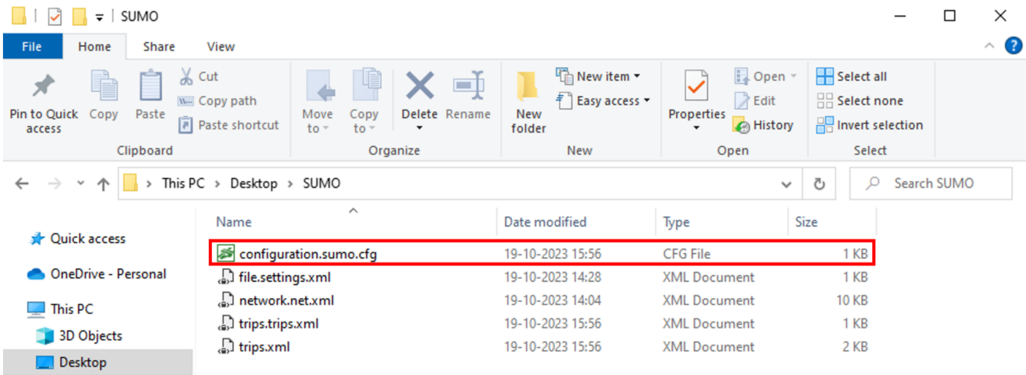


Figure 3-10: *Double click on Configuration.sumo.cfg*

- **Step 12:** Open Configuration.sumo.cfg by double clicking or open SUMO using sumo-gui.exe present in `<SUMO INSTALL DIRECTORY>/bin`. Open scenario in SUMO using Open → Simulation and verify whether the network loads and simulation happens as per the configuration done.

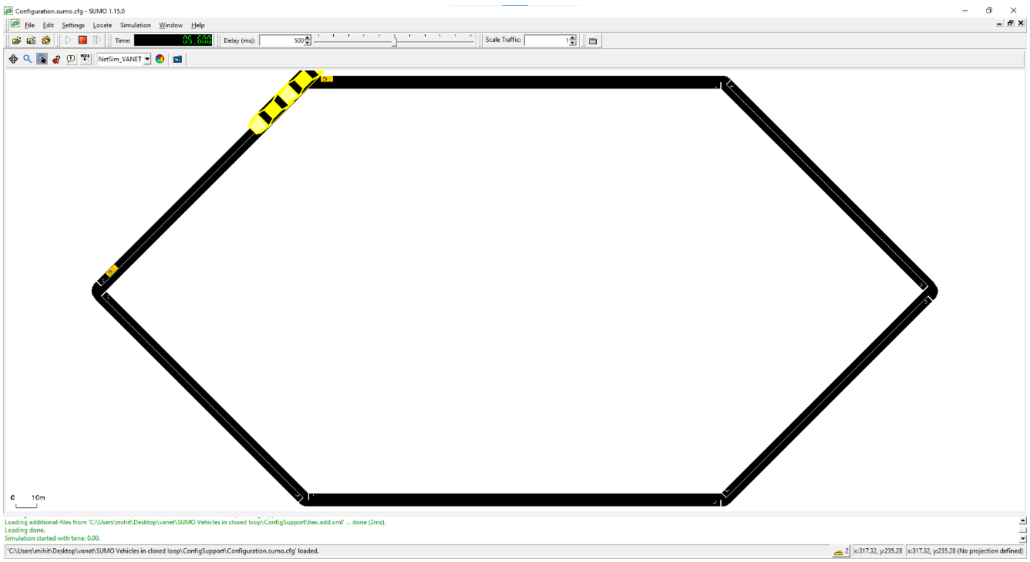


Figure 3-11: *SUMO simulation window*

- **Step 13:** Open the SUMO scenario via NetSim VANET by selecting VANET under the New Simulation in the NetSim Home Screen. Browse and locate the SUMO Configuration file present in your directory to load the road traffic network in NetSim GUI. The road network created in SUMO will be automatically replicated in NetSim GUI environment.

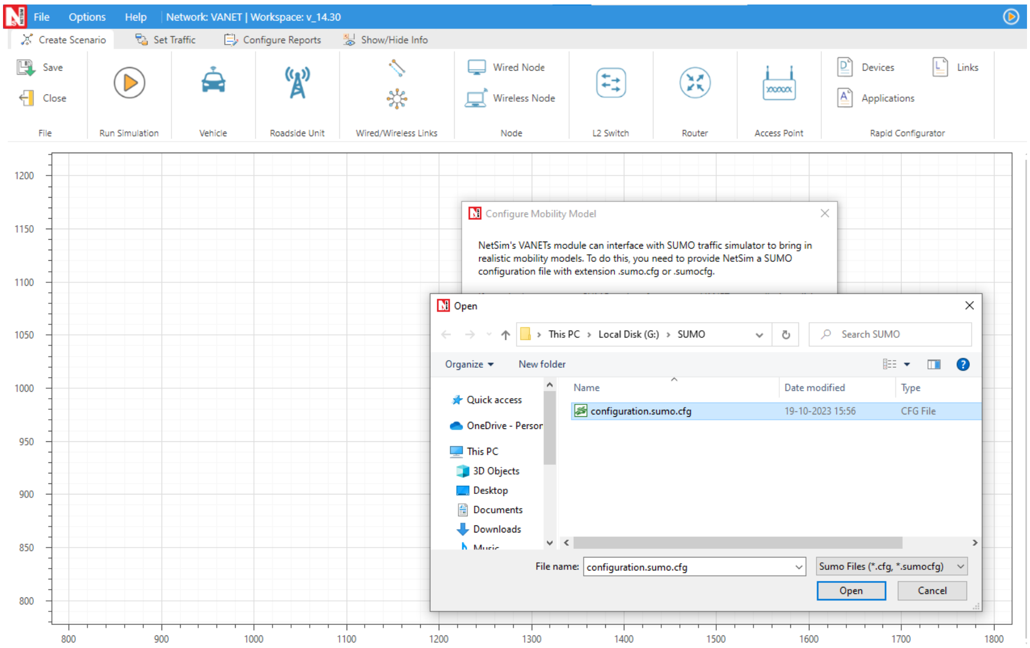


Figure 3-12: *Importing a sumo network configuration into NetSim*

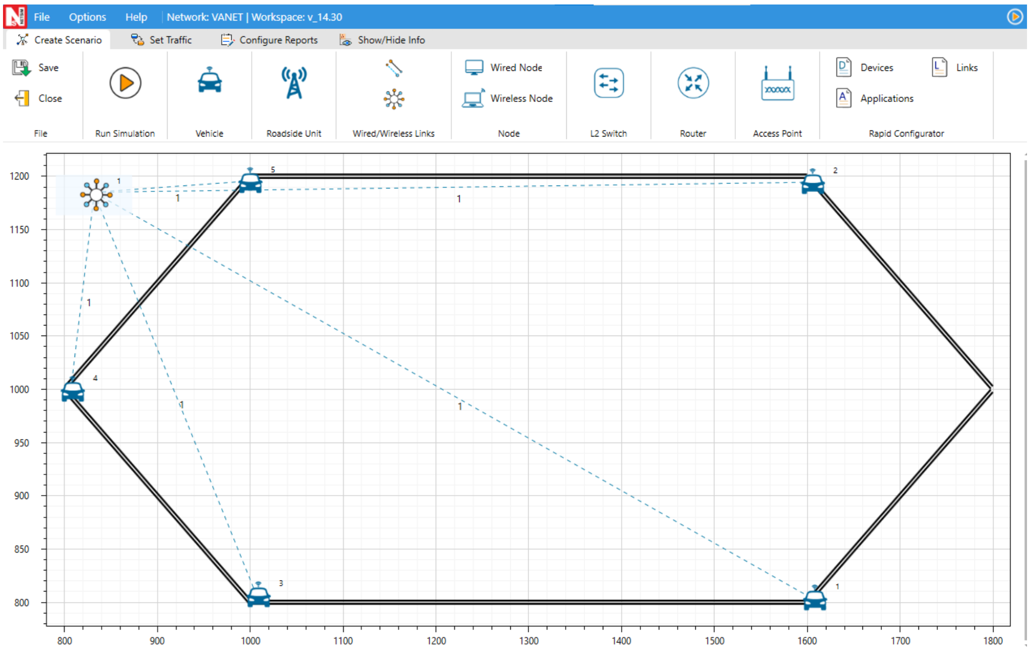


Figure 3-13: *Network Topology in this experiment*

- **Step 14:** Configure traffic between vehicles using the Application icon, enable trace files and plots.
- **Step 15:** Click on Run Simulation button. The Simulation Time is equal to the end time specified in sumo configuration (sumo.cfg) file and Simulation Time option is Non editable.

3.10.2 Creating your own network in SUMO manually

- **Step 1:** Create a node file using any code editor (like notepad, notepad++ etc.) and the file extension will be .nod.xml. It represents the junctions in the road. Each of these attributes has a certain meaning and value range: node id means unique name of each junction, x-y is the positions of node and type can be "priority", "traffic_light", "rail_crossing", "rail_signal" etc. (Refer: https://sumo.dlr.de/wiki/Networks/PlainXML#Node_Descriptions).

NODES.nod.xml

```xml
<nodes>
  <node id="n0" x="100" y="300" type="priority"/>
  <node id="n1" x="300" y="300" type="traffic_light"/>
  <node id="n2" x="500" y="300" type="rail_crossing"/>
  <node id="n3" x="700" y="300" type="priority"/>
  <node id="n4" x="300" y="100" type="traffic_light"/>
  <node id="n5" x="500" y="100" type="traffic_light"/>
</nodes>
```

Figure 3-14: *Device Positions in nodes file*

- **Step 2:** Create an edge file that describes how the junctions or nodes are connected to each other. The extension of this file is .edg.xml. Each edge is unidirectional and starts at the "from"-node and ends at the "to"-node. For each edge, some further attributes should be supplied, being the number of lanes the edge has (numLanes), the maximum speed allowed on the edge speed. Furthermore, the priority may be defined optionally. (Refer: https://sumo.dlr.de/wiki/Networks/PlainXML#Edge_Descriptions).

EDGE.edg.xml

```xml
<edges>
<edge id="L01" from="n1" to="n0" priority="75" nolanes="2" speed="40"/>
<edge id="R01" from="n0" to="n1" priority="75" nolanes="2" speed="40"/>
<edge id="L12" from="n2" to="n1" priority="75" nolanes="2" speed="40"/>
<edge id="R12" from="n1" to="n2" priority="75" nolanes="2" speed="40"/>
<edge id="L23" from="n3" to="n2" priority="75" nolanes="2" speed="40"/>
<edge id="R23" from="n2" to="n3" priority="75" nolanes="2" speed="40"/>
<edge id="D04" from="n0" to="n4" priority="75" nolanes="2" speed="40"/>
<edge id="U04" from="n4" to="n0" priority="75" nolanes="2" speed="40"/>
<edge id="D14" from="n1" to="n4" priority="75" nolanes="2" speed="40"/>
<edge id="U14" from="n4" to="n1" priority="75" nolanes="2" speed="40"/>
<edge id="D35" from="n3" to="n5" priority="75" nolanes="2" speed="40"/>
<edge id="U35" from="n5" to="n3" priority="75" nolanes="2" speed="40"/>
<edge id="D25" from="n2" to="n5" priority="75" nolanes="2" speed="40"/>
<edge id="U25" from="n5" to="n2" priority="75" nolanes="2" speed="40"/>
<edge id="L45" from="n5" to="n4" priority="75" nolanes="2" speed="40"/>
<edge id="R45" from="n4" to="n5" priority="75" nolanes="2" speed="40"/>
</edges>
```

Figure 3-15: *Edge file*

- **Step 3:** Open Command Prompt and change the directory to the binary folder of sumo using cd command. `"cd C:\Program Files (x86)\Eclipse\Sumo\bin"`

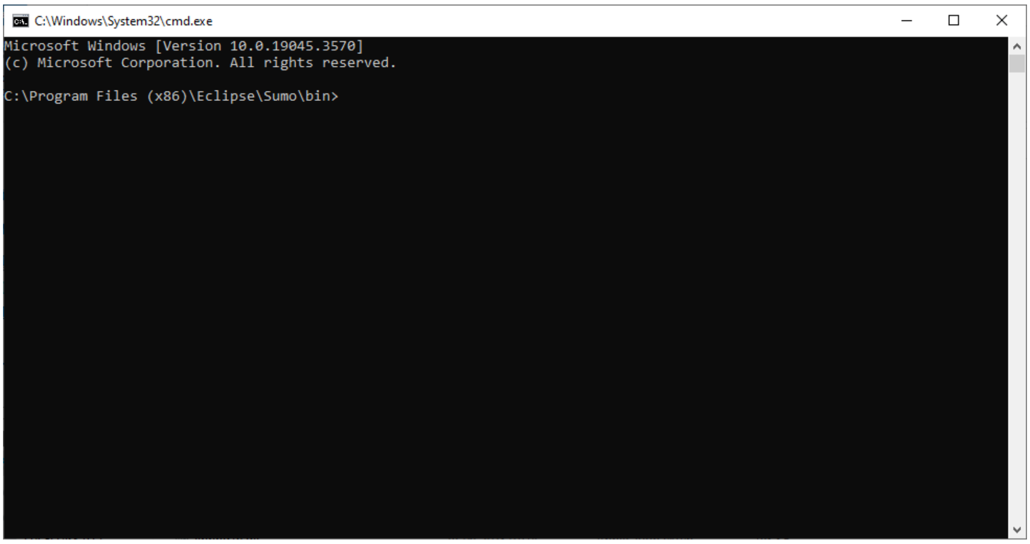


Figure 3-16: *Open command prompt in installation directory*

- **Step 4:** Generate Network file by using NETCONVERT command. Make a folder named like VANET_Example and place the .nod.xml and .edg.xml files i.e. NODES.nod.xml and EDGE.edg.xml respectively.

```
netconvert --n "<path>\<filename>.nod.xml"
           --e "<path>\<filename>.edg.xml"
           --o "<path>\<filename>.net.xml"
```

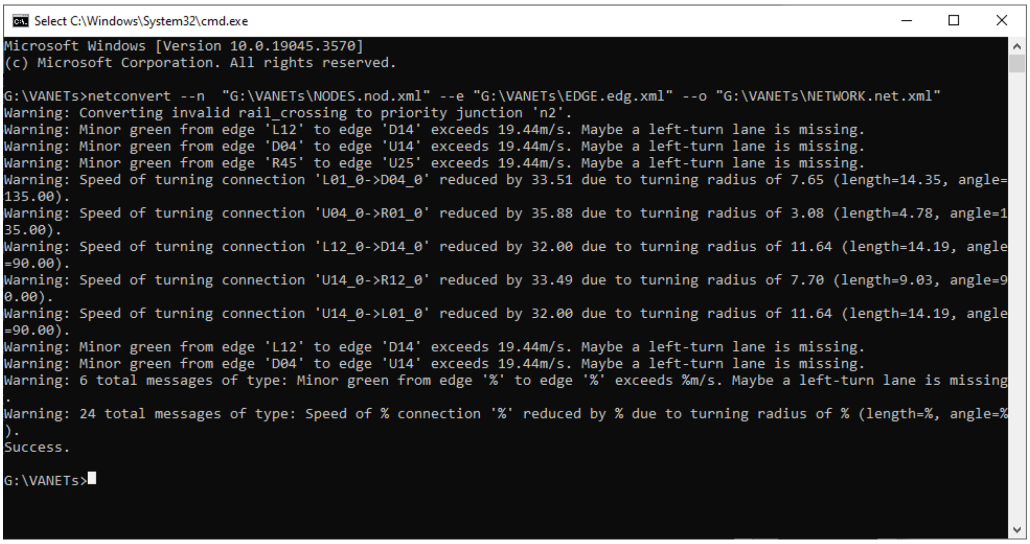


Figure 3-17: *Generating Network file by using NETCONVERT command*

- **Step 5:** Create a .rou.xml file that describes the direction of the vehicle's movement.

VEHICLES.rou.xml

```
<routes>
<!-- e -->
<vehicle id="V0" depart="0.00">
<route edges="D04 R45 U35 L23 L12 L01" />
</vehicle>
<vehicle id="V1" depart="0.00">
<route edges="R12 R23 D35 L45 U14 L01" />
</vehicle>
<vehicle id="V2" depart="0.00">
<route edges="R01 R12 R23 D35 L45 U04" />
</vehicle>
<vehicle id="V3" depart="0.00">
<route edges="R01 D14 R45 U35 L23 L12" />
</vehicle>
<vehicle id="V4" depart="0.00">
<route edges="D35 L45 U14 R12 D25 U35 L23" />
</vehicle>
<vehicle id="V5" depart="0.00">
<route edges="R45 U25 L12 L01 D04 R45" />
</vehicle>
</routes>
```

Figure 3-18: *Direction of the vehicle's movement*

- **Step 6:** Create a sumo configuration file using any code editor (like notepad, notepad++ etc.) and the extension is .sumo.cfg. Place the file inside the same folder where the network file (i.e. NETWORK.net.xml) and route file (i.e. VEHICLES.rou.xml) are present.

configuration.sumo.cfg

```
<Configuration>
<input>
<net-file value="NETWORK.net.xml"/>
<route-files value="VEHICLES.rou.xml"/>
</input>
<time>
<begin value="0"/>
<end value="200"/>
<step-length value="1"/>
</time>
</Configuration>
```

Figure 3-19: *Sumo configuration file*

- **Step 7:** Now open "New Simulation → VANET". Choose the Configuration.sumo.cfg from the specified folder and run simulation using NetSim.

3.11 How to include Roadside Units (RSUs) in a VANET network?

Upon importing a sumo network configuration into NetSim, roads and vehicles are automatically added in NetSim as per the configuration done in SUMO.

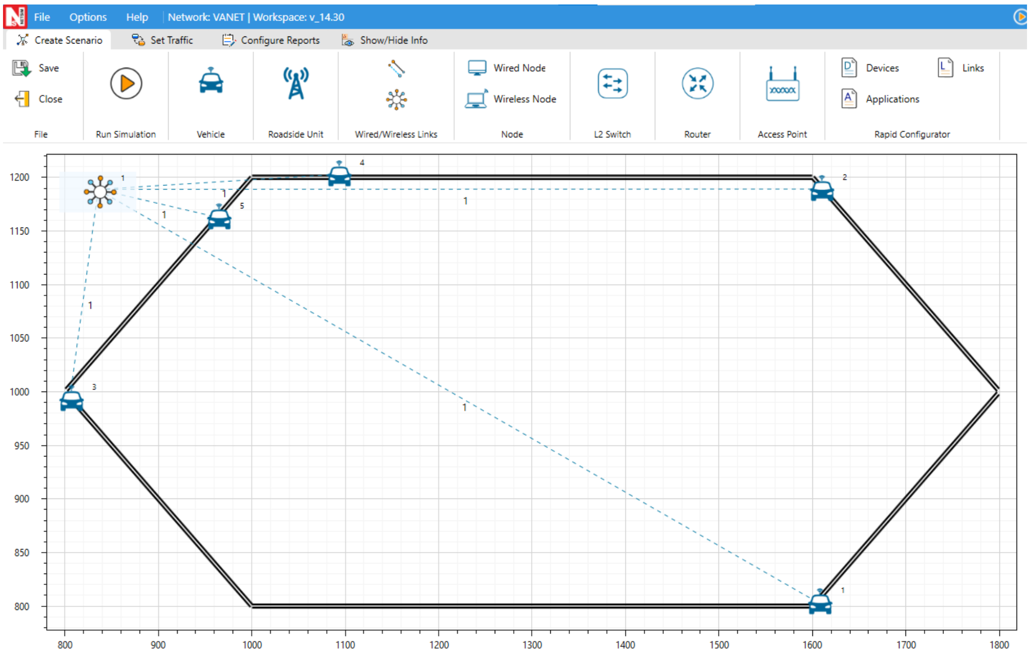


Figure 3-20: *Importing a sumo network configuration into NetSim*

Roadside Units can be optionally included in the network by manually clicking and dropping the RSU icon from the ribbon.

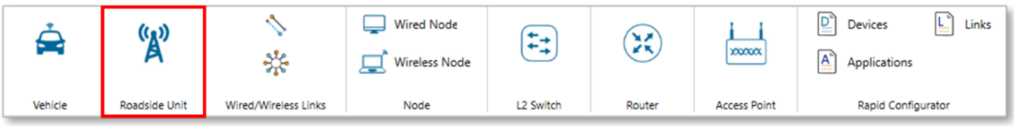


Figure 3-21: *RSU icon from the ribbon*

RSUs should be connected using ad hoc links manually.

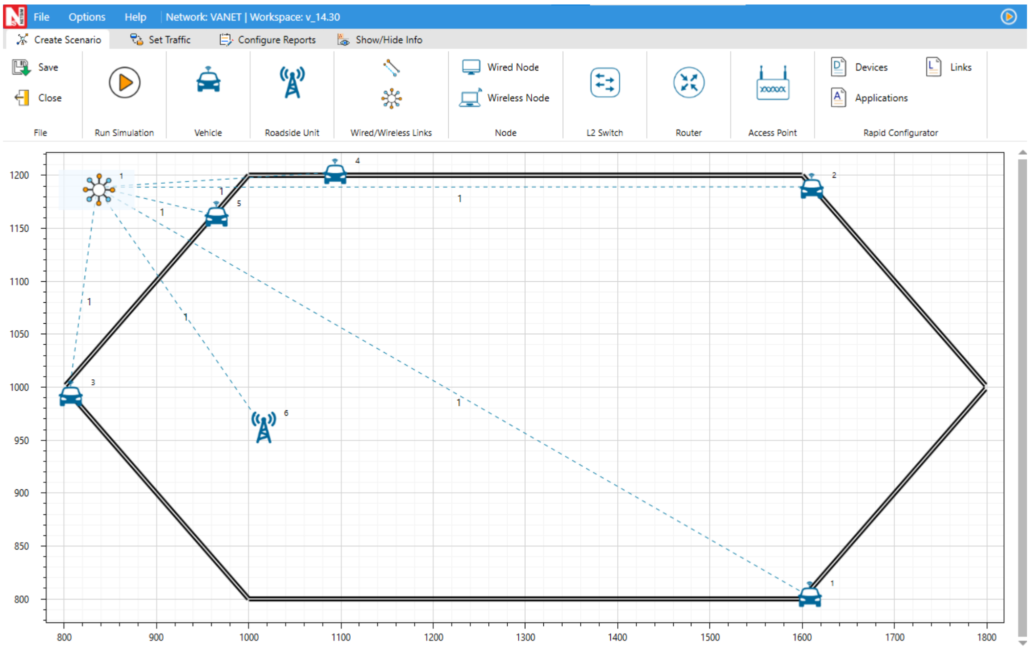


Figure 3-22: *NetSim Design Window*

Traffic can be configured between RSUs and vehicles via Application configuration.

3.12 Radio measurement log file and plots

NetSim IEEE 802.11 Radio Measurements log file records pathloss, shadowing loss, fading loss, received power, transmitted power, SNR, and BER. This log can be enabled by clicking on configure reports in top ribbon ¿ Plots ¿ Select IEEE 802.11 Radio Measurements log under Network Logs as shown below.

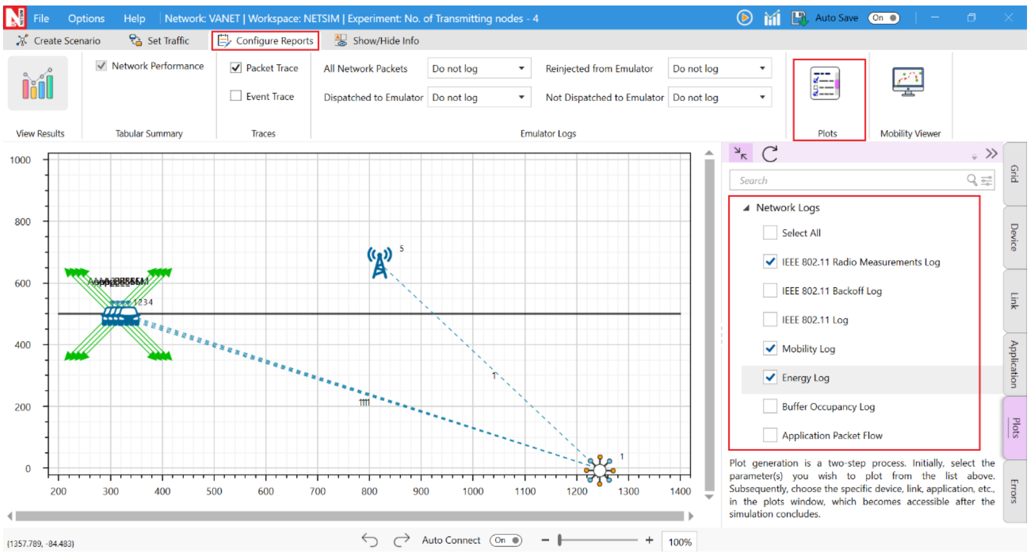


Figure 3-23: *Enabling IEEE 802.11 Radio Measurements log for VANETs*

The IEEE802.11 Radio Measurements log.csv file will contain the following information:

- Time in Milliseconds
- Transmitter Name
- Receiver Name

- Distance between the Transmitter and the Receiver in meters
- Packet ID
- Packet Type
- Control Packet Type
- Transmitter Power in dBm
- Pathloss in dB
- Shadowing Loss in dB
- Fading Loss in dB
- Total Loss in dB
- Transmitter Antenna Gain in dB
- Receiver Antenna Gain in dB
- Received Power in dBm
- Interference in dBm
- SNR in dB
- SINR in dB
- NSS (Number of Spatial Streams)
- BER
- MCS Index

The log file can be accessed from the Simulations Results Window under the logs as shown below.

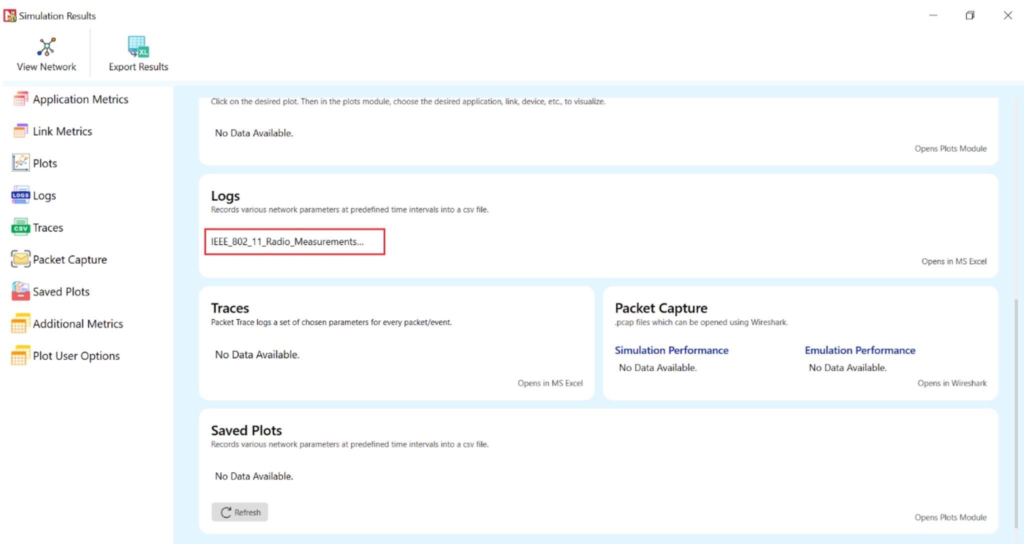
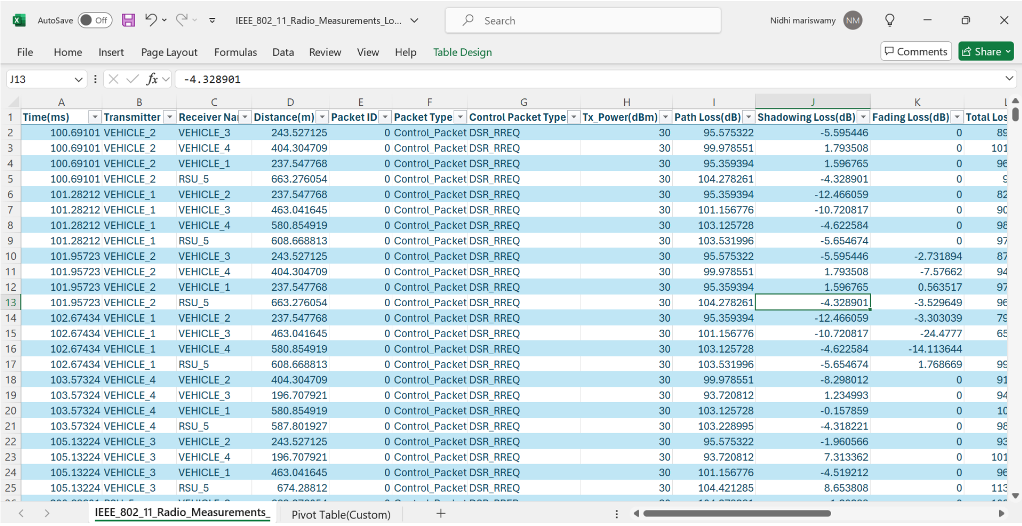


Figure 3-24: *Opening IEEE 802.11 Radio Measurements log from simulation results window.*



	Time(ms)	Transmitter	Receiver Na	Distance(m)	Packet ID	Packet Type	Control Packet Type	Tx_Power(dBm)	Path Loss(dB)	Shadowing Loss(dB)	Fading Loss(dB)	Total Los
2	100.69101	VEHICLE_2	VEHICLE_3	243.527125	0	Control_Packet	DSR_RREQ	30	95.575322	-5.595446	0	89
3	100.69101	VEHICLE_2	VEHICLE_4	404.304709	0	Control_Packet	DSR_RREQ	30	99.978551	1.793508	0	101
4	100.69101	VEHICLE_2	VEHICLE_1	237.547768	0	Control_Packet	DSR_RREQ	30	95.359394	1.596765	0	96
5	100.69101	VEHICLE_2	RSU_5	663.276054	0	Control_Packet	DSR_RREQ	30	104.278261	-4.328901	0	9
6	101.28212	VEHICLE_1	VEHICLE_2	237.547768	0	Control_Packet	DSR_RREQ	30	95.359394	-12.466059	0	82
7	101.28212	VEHICLE_1	VEHICLE_3	463.041645	0	Control_Packet	DSR_RREQ	30	101.156776	-10.720817	0	90
8	101.28212	VEHICLE_1	VEHICLE_4	580.854919	0	Control_Packet	DSR_RREQ	30	103.125728	-4.622584	0	98
9	101.28212	VEHICLE_1	RSU_5	608.668813	0	Control_Packet	DSR_RREQ	30	103.531996	-5.654674	0	97
10	101.95723	VEHICLE_2	VEHICLE_3	243.527125	0	Control_Packet	DSR_RREQ	30	95.575322	-5.595446	-2.731894	87
11	101.95723	VEHICLE_2	VEHICLE_4	404.304709	0	Control_Packet	DSR_RREQ	30	99.978551	1.793508	-7.57662	94
12	101.95723	VEHICLE_2	VEHICLE_1	237.547768	0	Control_Packet	DSR_RREQ	30	95.359394	1.596765	0.563517	97
13	101.95723	VEHICLE_2	RSU_5	663.276054	0	Control_Packet	DSR_RREQ	30	104.278261	-4.328901	-3.529649	96
14	102.67434	VEHICLE_1	VEHICLE_2	237.547768	0	Control_Packet	DSR_RREQ	30	95.359394	-12.466059	-3.303039	79
15	102.67434	VEHICLE_1	VEHICLE_3	463.041645	0	Control_Packet	DSR_RREQ	30	101.156776	-10.720817	-24.4777	65
16	102.67434	VEHICLE_1	VEHICLE_4	580.854919	0	Control_Packet	DSR_RREQ	30	103.125728	-4.622584	-14.113644	
17	102.67434	VEHICLE_1	RSU_5	608.668813	0	Control_Packet	DSR_RREQ	30	103.531996	-5.654674	1.768669	99
18	103.57324	VEHICLE_4	VEHICLE_2	404.304709	0	Control_Packet	DSR_RREQ	30	99.978551	-8.298012	0	91
19	103.57324	VEHICLE_4	VEHICLE_3	196.707921	0	Control_Packet	DSR_RREQ	30	93.720812	1.234993	0	94
20	103.57324	VEHICLE_4	VEHICLE_1	580.854919	0	Control_Packet	DSR_RREQ	30	103.125728	-0.157859	0	10
21	103.57324	VEHICLE_4	RSU_5	587.801927	0	Control_Packet	DSR_RREQ	30	103.228995	-4.318221	0	98
22	105.13224	VEHICLE_3	VEHICLE_2	243.527125	0	Control_Packet	DSR_RREQ	30	95.575322	-1.960566	0	93
23	105.13224	VEHICLE_3	VEHICLE_4	196.707921	0	Control_Packet	DSR_RREQ	30	93.720812	7.313362	0	101
24	105.13224	VEHICLE_3	VEHICLE_1	463.041645	0	Control_Packet	DSR_RREQ	30	101.156776	-4.519212	0	96
25	105.13224	VEHICLE_3	RSU_5	674.28812	0	Control_Packet	DSR_RREQ	30	104.421285	8.653808	0	113

Figure 3-25: *IEEE 802.11 Radio Measurements log file.*

Enabling the plots is explained in the section 2.5. Here is the plot showing the transmit energy vs. time for a DSR scenario with five vehicles.

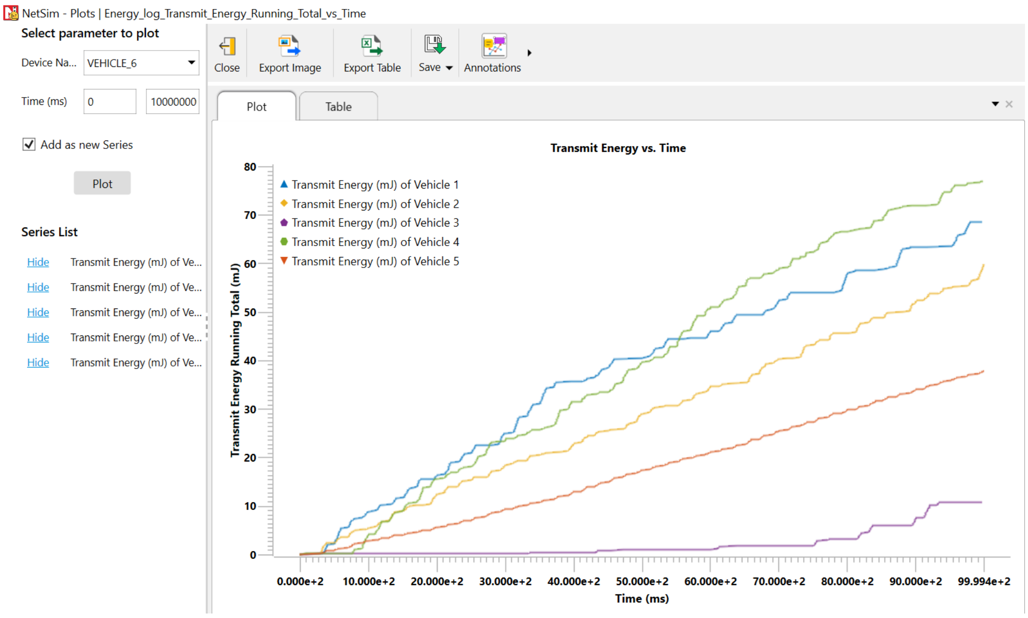


Figure 3-26: *Plot of Transmit Energy vs Time for VANET*

4 Featured Examples

- Sample configuration files for all networks are available in Examples Menu in NetSim Home Screen. These files provide examples on how NetSim can be used – the parameters that can be changed and the typical effect it has on performance.

NOTE: In all VANET featured examples, the error model is set to SINR BER by Formula.

4.1 Importing a simple VANET scenario from SUMO

Open NetSim and Select Examples ¿ VANETs ¿ Importing a simple VANET scenario from SUMO then click on the tile in the middle panel to load the example as shown in below screenshot.

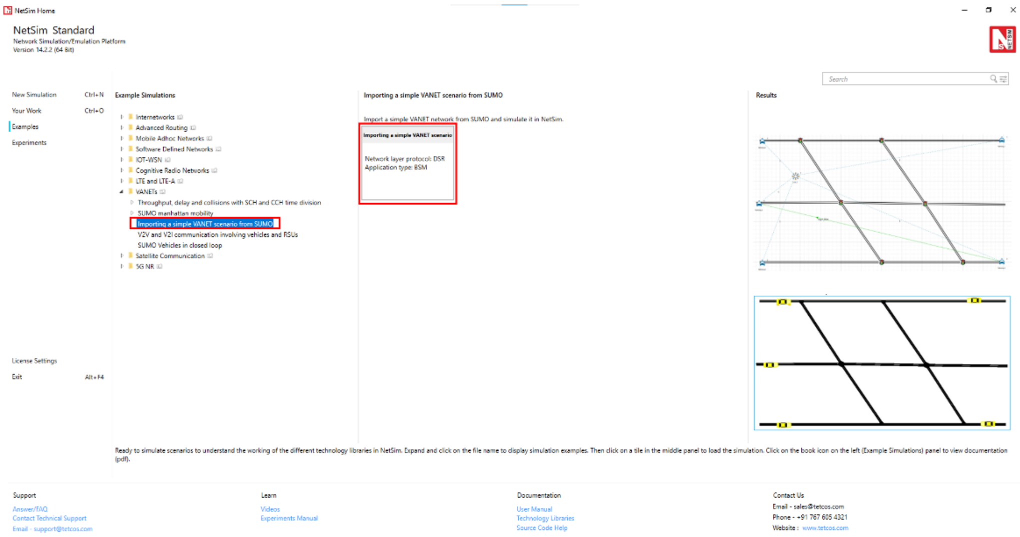


Figure 4-1: *List of scenarios for the example of Importing a simple VANET scenario from SUMO*

This scenario involves a simple road traffic network scenario as shown below:

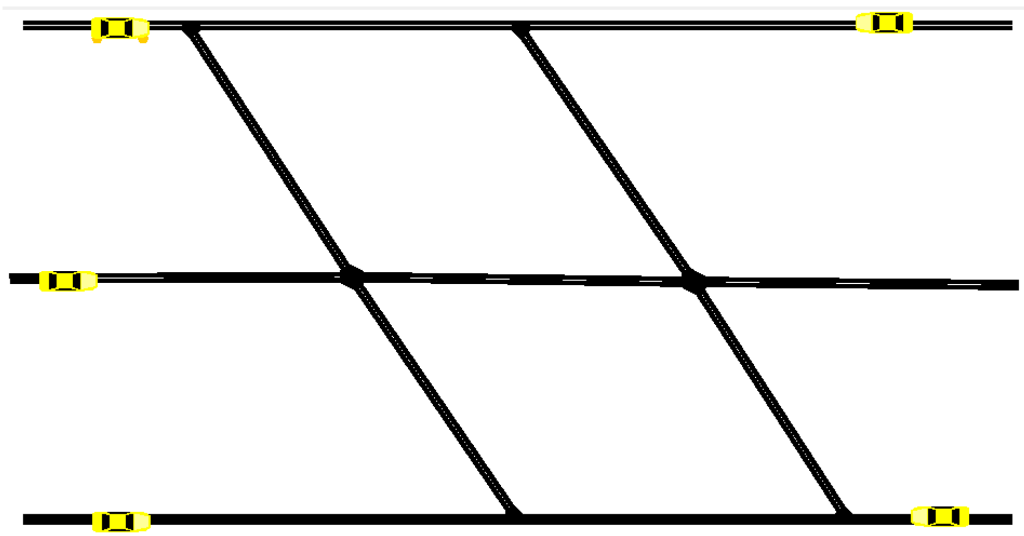


Figure 4-2: *Network topology in Sumo*

An equivalent network is created in NetSim by importing the SUMO configuration file. In NetSim the TCP/IP stack parameters of the devices are configured along with network traffic between vehicles.

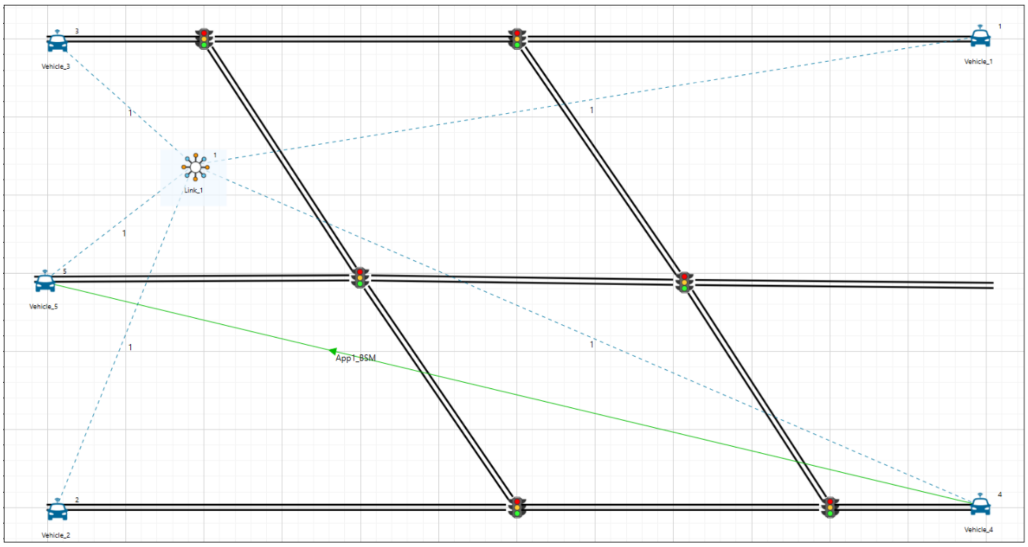


Figure 4-3: *Network scenario after importing into NetSim and configuring application traffic*

The properties of Application, vehicle and link are set as follows:

Table 4-1: *Application, Link and Physical layer Properties*

Properties	Properties
Application Properties	
Application Type	BSM
Application Method	UNICAST
Packet Size	20 (Bytes)
Inter Arrival Time	1000000 (μs)
Link Properties	
Channel Characteristics	No Pathloss
Physical Layer Properties (Vehicle)	
Standard	IEEE802.11p
Transmitter Power	40 mW
Antenna Gain	1 dBi
Antenna Height	1 m
Bandwidth	10 MHz

Note that the packet trace is enabled under the Configure Reports tab, and the mobility log is enabled under the Network Logs present in the plots on the right side. This allows for the recording of data traffic flow and vehicular movement.

After running the simulation, Packet Trace can be used to visualize packet flow along with packet information and Mobility log can be used to record vehicle mobility.

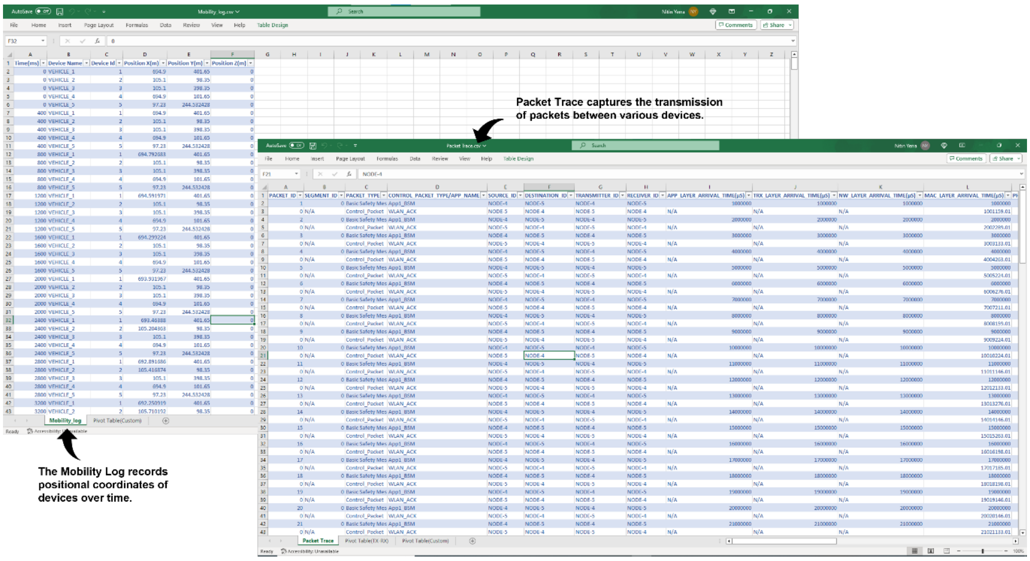


Figure 4-4: *Packet Trace and Mobility log window*

Simulation results dashboard provides the performance metrics of protocols running in different layers of the network stack of the devices.

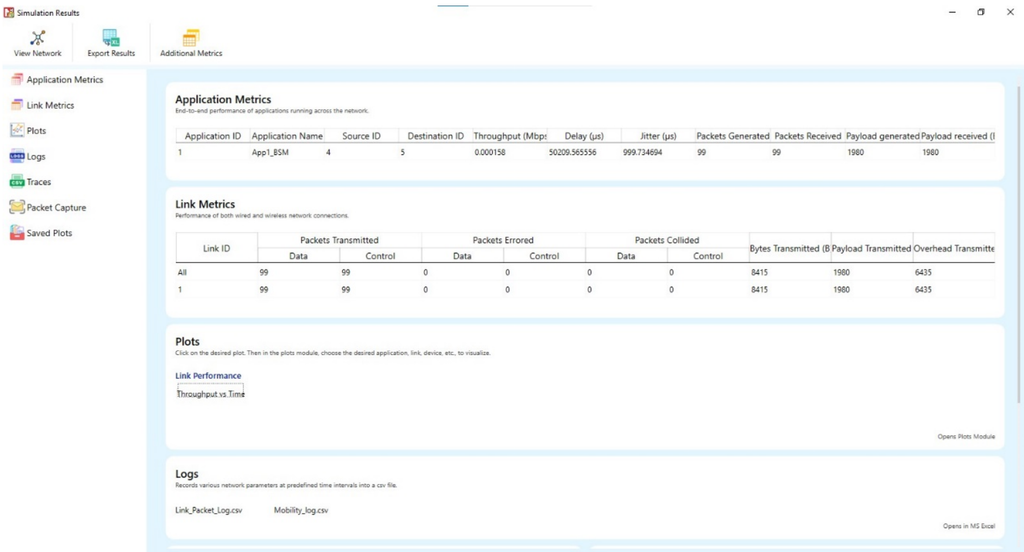


Figure 4-5: *Result Dashboard*

4.2 V2V and V2I communication involving Vehicles and RSUs

Open NetSim and Select Examples ¿ VANETs ¿ V2V and V2I communication involving Vehicles and RSUs then click on the tile in the middle panel to load the example as shown in below screenshot.

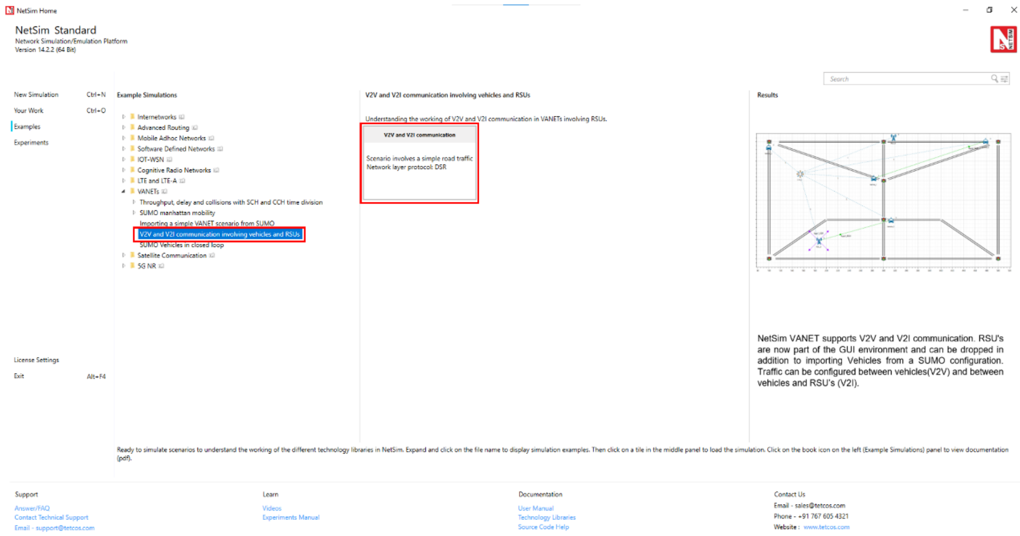


Figure 4-6: *List of scenarios for the example of V2V and V2I communication involving Vehicles and RSUs*

NetSim VANETs module supports V2V and V2I communication. RSUs are now part of the GUI environment and can be dropped in addition to importing Vehicles from a SUMO configuration. Traffic can be configured between vehicles (V2V) and between vehicles and RSUs (V2I).

This scenario involves a simple road traffic network scenario as shown below:

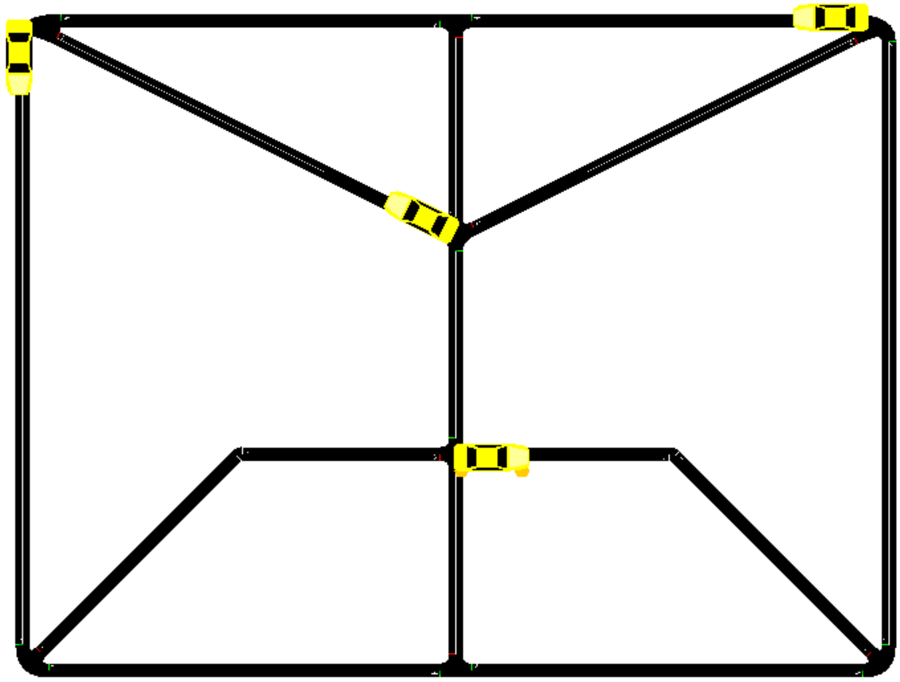


Figure 4-7: *Network topology in Sumo*

An equivalent network is created in NetSim by importing the SUMO configuration file as shown below:

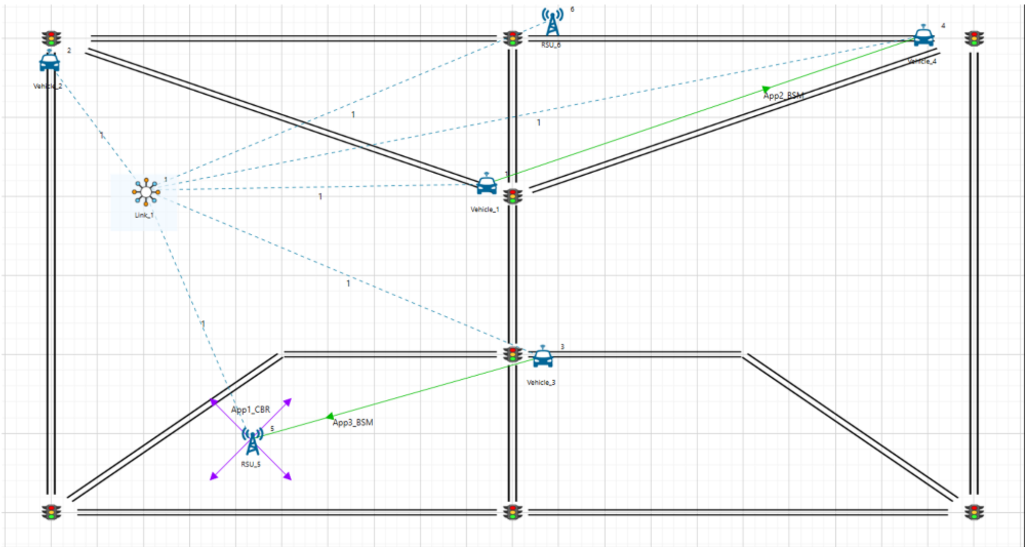


Figure 4-8: *Network set up for scenario involving vehicles and RSUs for V2V and V2I communication*

After importing the SUMO configuration file in NetSim, RSUs were added at junctions. In NetSim the TCP/IP stack parameters of the devices are configured along with network traffic between vehicles and between vehicles and RSUs.

Settings done for the Experiment:

Table 4-2: *Application, Link and Physical layer Properties*

Properties	**Properties**
Application-1 Properties	
App Type	CBR
Application Method	BROADCAST
Transport Protocol	UDP
Packet Size	1460 (Bytes)
Inter Arrival Time	1000000 (μs)
Application-2 and 3 Properties	
App Type	BSM
Application Method	UNICAST
Packet Size	20 (Bytes)
Inter Arrival Time	1000000 (μs)
Link Properties	
Channel Characteristics	No Pathloss
Physical Layer Properties (Vehicles & RSU)	
Standard	IEEE802.11p
Transmitter Power	40 mW
Antenna Gain	1 dBi
Antenna Height	1 m
Bandwidth	10 MHz

Note that the packet trace is enabled under the Configure Reports tab, and the mobility log is enabled under the Network Logs present in the plots on the right side. This allows for the recording of data traffic flow and vehicular movement.

After running the simulation, Packet Trace can be used to visualize packet flow along with packet information and mobility log can be used to record vehicle mobility.

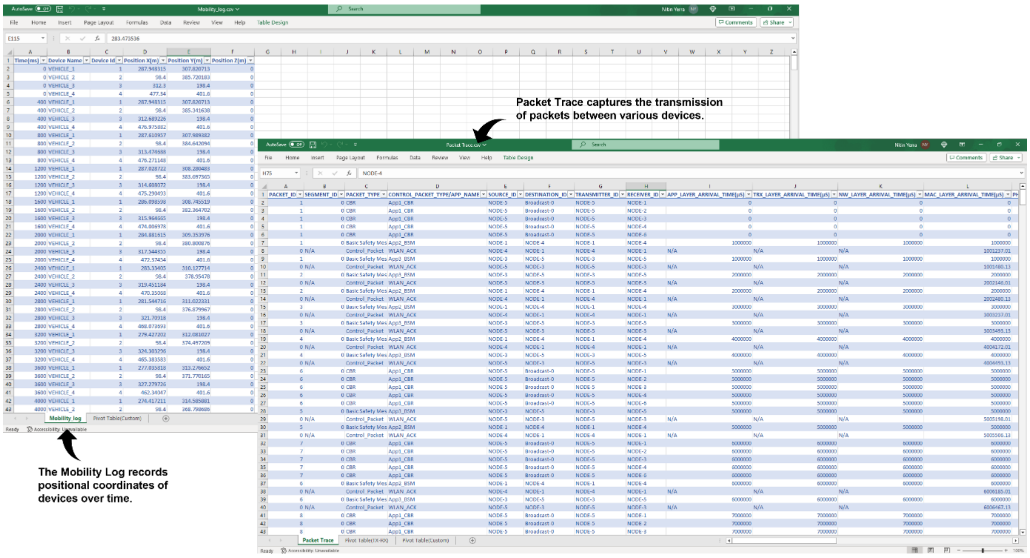


Figure 4-9: *Packet Trace and Mobility log window*

Simulation results dashboard provides the performance metrics of protocols running in different layers of the network stack of the devices.

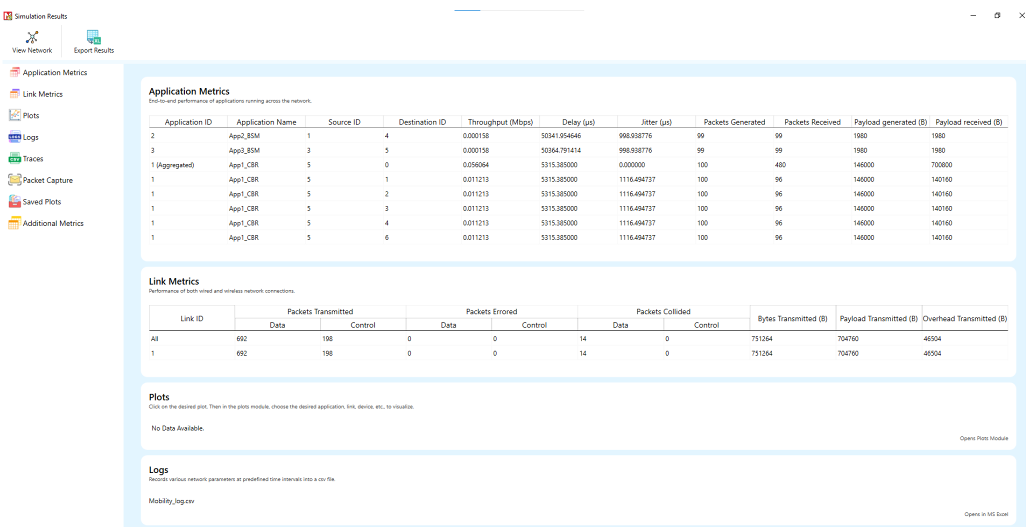


Figure 4-10: *Result Dashboard*

4.3 Throughput, delay and collisions with SCH and CCH time division

All the following examples are available in NetSim GUI. Navigate to Example ¿ VANETs ¿ Throughput, delay and collisions with SCH and CCH time division. Within Throughput, delay and collisions with SCH and CCH time division users will see four folders. Each folder comprises simulation samples for Parts 1, 2, 3 and 4 as explained below.

4.3.1 Background

Dedicated short range communication (DSRC) which uses two channels: Service channel SCH and Control channel (CCH). Each synchronization interval SI is split as follows:

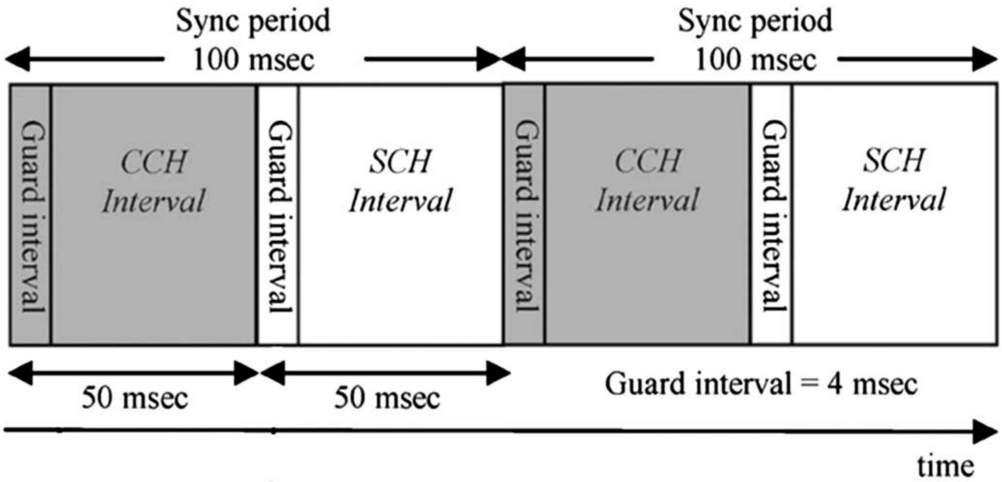


Figure 4-11: *We see the time divisions in DSRC. Each synchronization period consists of 1 CCH, 1 SCH and 1 guard interval. While the sync period is generally equal to 100 ms. NetSim allows users to modify the CCH and SCH interval, and in turn the total Sync period.*

All devices switch between SCH and CCH and the alternation is based on the time divisions. NetSim allows the user to configure values of CCH interval, SCH interval and Guard interval. The default channels used in NetSim are SCH 171 (5.855 GHz) and CCH 178 (5.890 GHz).

Multiple nodes access the medium using 802.11p protocol. IEEE 802.11p PHY operates in the 5.9 GHz band with a channel bandwidth of 10 MHz. 802.11p is an adaptation of the IEEE 802.11a standard used in Wi-Fi systems.

4.3.2 Simulation scenario

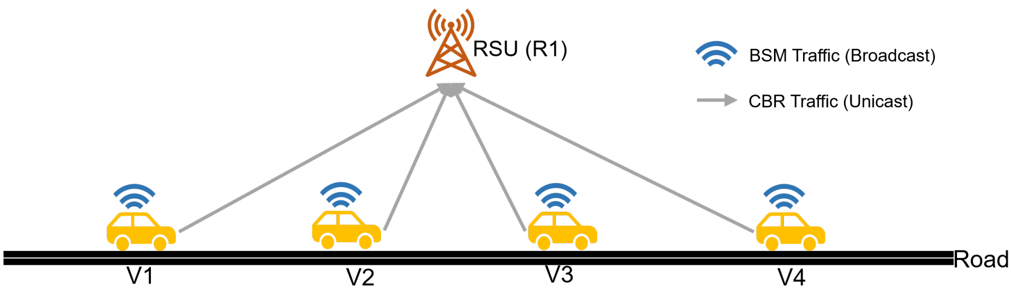


Figure 4-12: *Illustration of the VANET scenario under study. The network comprises 4 vehicles and 1 roadside unit. Each vehicle transmits two applications: (i) a BSM broadcast application that is sent to all other devices (vehicles plus RSU) within range and (ii) a CBR application transmitted to the RSU*

The scenario comprises four vehicles, V1, V2, V3 and V4 communicating to the RSU, R1 and to one another. Each vehicle sends unicast CBR traffic to the RSU and broadcasts BSM (safety messages) to one another. Recall that per DSRC functioning, BSM is sent on the CCH while CBR is sent on the SCH.

4.3.3 Simulation parameters and results

4.3.4 Part 1: Throughput

Open NetSim and Select Examples ¿ VANETs ¿ Throughput, delay and collisions with SCH and CCH time division ¿ Throughput then click on the tile in the middle panel to load the example as shown in below figure.

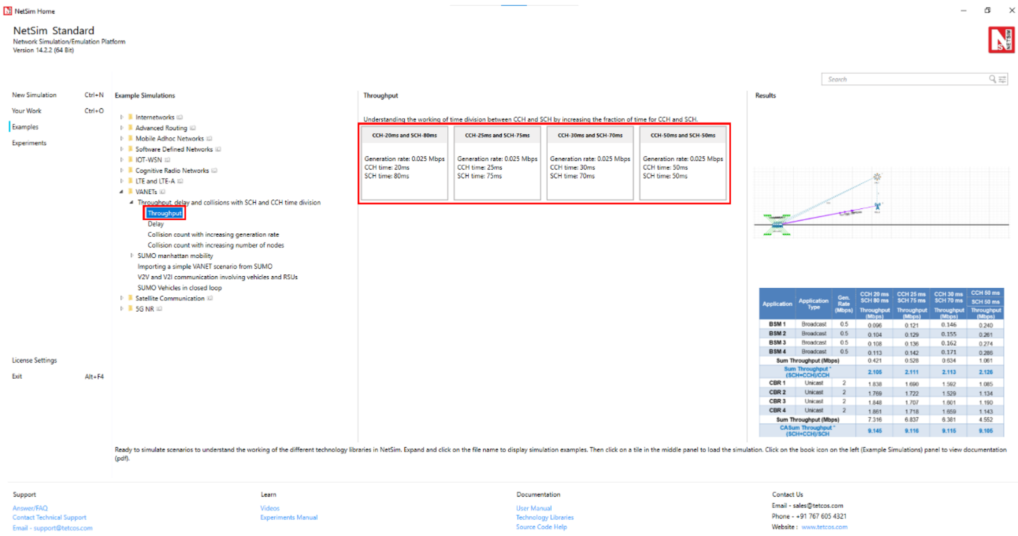


Figure 4-13: *List of scenarios for the example of Throughput*

The following network diagram illustrates what the NetSim UI displays when you open the example configuration file throughput as shown in below figure.

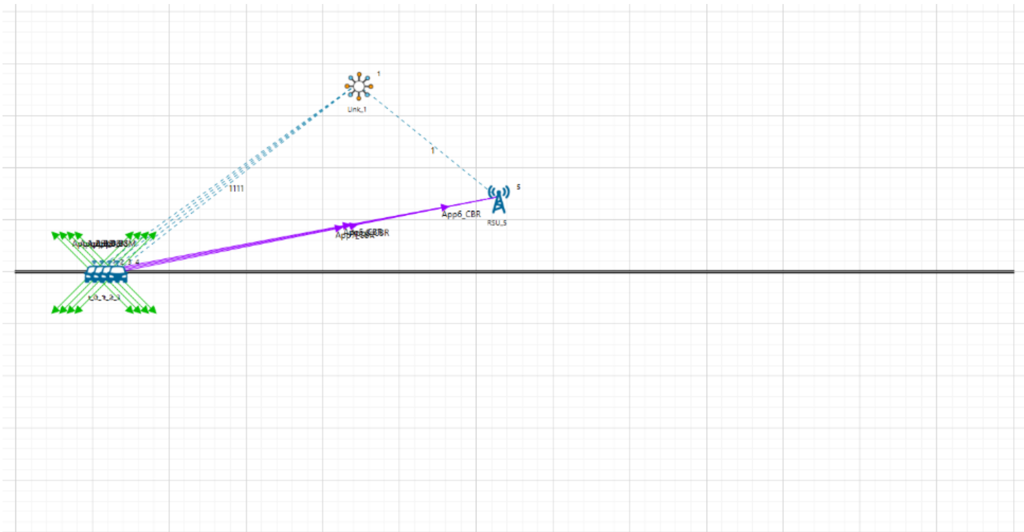


Figure 4-14: *Network setup for studying the Throughput*

Settings done for the Experiment:

Table 4-3: *Application, link and Physical layer Properties*

Properties	Properties
Application Properties	
Application Type	BSM (Applications 1–4)
Application Method	Broadcast
Packet Size	20 (Bytes)
Inter Arrival Time	320 (μs)
Application Type	CBR (Applications 5–8)
Application Method	Unicast
Packet Size	1460 (Bytes)
Inter Arrival Time	6147.4 (μs)
Link Properties	
Channel Characteristics	Pathloss Only
Pathloss Model	Log distance
Pathloss Exponent	2
Datalink Properties	
CCH Time	20 ms, 25 ms, 30 ms, 50 ms (Varying)
SCH Time	80 ms, 75 ms, 70 ms, 50 ms (Varying)
Physical Layer Properties (Vehicle)	
Standard	IEEE802.11p
Transmitter Power	100 mW
Antenna Gain	1 dBi
Antenna Height	1 m
Bandwidth	10 MHz

Note that the packet trace is enabled under the Configure Reports tab, and the mobility log is enabled under the Network Logs present in the plots on the right side. This allows for the recording of data traffic flow and vehicular movement.

Results The BSM application is configured with packet size of 20B and inter-packet arrival time of 320 μs, while the CBR application is configured with packet size of 1460B and inter-packet arrival time of 5840 μs.

Table 4-4: *We see that as the CCH interval increases, BSM application has higher throughput rate. Similarly, as the SCH Interval decreases there is a decrease in throughput rate.*

App	App Type	Gen. Rate (Mbps)	CCH 20ms SCH 80ms	CCH 25ms SCH 75ms	CCH 30ms SCH 70ms	CCH 50ms SCH 50ms
			Throughput (Mbps)			
BSM 1	Broadcast	0.5	0.096	0.122	0.144	0.241
BSM 2	Broadcast	0.5	0.103	0.129	0.156	0.261
BSM 3	Broadcast	0.5	0.109	0.135	0.163	0.274
BSM 4	Broadcast	0.5	0.113	0.141	0.171	0.285
Sum Throughput (Mbps)			0.421	0.527	0.634	1.061

App	App Type	Gen. Rate (Mbps)	CCH 20ms SCH 80ms	CCH 25ms SCH 75ms	CCH 30ms SCH 70ms	CCH 50ms SCH 50ms
Sum Throughput × (SCH+CCH)/CCH			2.106	2.107	2.114	2.123
CBR 1	Unicast	1.9	1.826	1.726	1.544	1.139
CBR 2	Unicast	1.9	1.778	1.691	1.604	1.190
CBR 3	Unicast	1.9	1.837	1.668	1.622	1.119
CBR 4	Unicast	1.9	1.884	1.762	1.619	1.087
Sum Throughput (Mbps)			7.328	6.849	6.391	4.537
Sum Throughput × (SCH+CCH)/SCH			9.160	9.132	9.130	9.075

Observations

1. BSM is sent on CCH; CBR is sent on SCH. Increasing the fraction of time for CCH increases BSM throughput. Increasing the fraction of time for SCH increases CBR throughput.
2. As expected, Sum throughput divided by SCH fraction is equal for all cases. Similarly, Sum throughput divided by CCH fraction is equal in all cases. This verifies the working of time division between CCH and SCH.

4.3.5 Part 2: Delay

Open NetSim and Select Examples ¿ VANETs ¿ Throughput, delay and collisions with SCH and CCH time division ¿ Delay then click on the tile in the middle panel to load the example as shown in below screenshot.

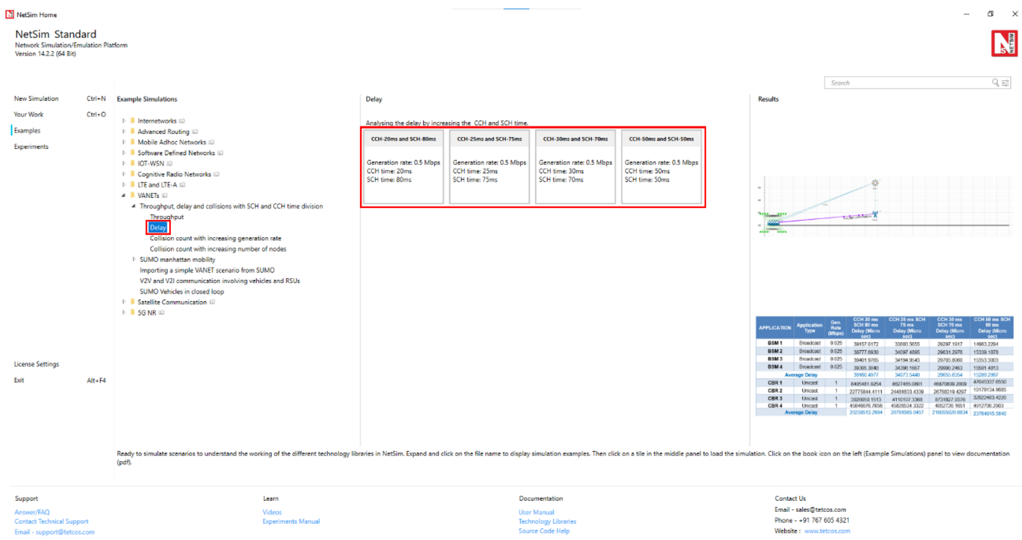


Figure 4-15: *List of scenarios for the example of Delay*

The following network diagram illustrates what the NetSim UI displays when you open the example configuration file as shown in below figure.

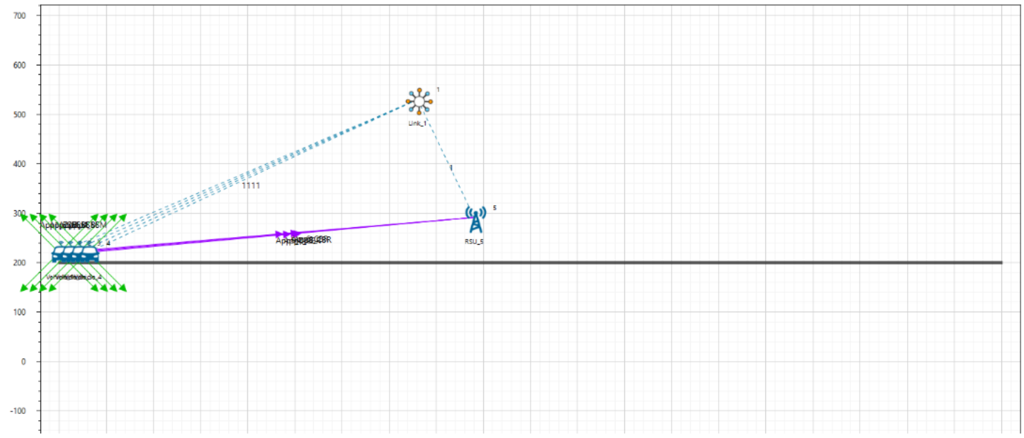


Figure 4-16: *Network setup for studying the Delay*

Settings done for the Experiment:

Table 4-5: *Application, Link and Physical layer Properties*

Properties	Properties
Application Properties	
Application Type	BSM (Applications 1–4)
Application Method	BROADCAST
Transport Protocol	WSMP
Packet Size	20 (Bytes)
Inter Arrival Time	6400 (μs)
Application Type	CBR (Applications 5–8)
Application Method	UNICAST
Transport Protocol	UDP
Packet Size	1460 (Bytes)
Inter Arrival Time	11680 (μs)
Link Properties	
Channel Characteristics	Pathloss Only
Pathloss Model	Log Distance
Pathloss Exponent	2.5
Data Link Properties	
CCH Time	20 ms, 25 ms, 30 ms, 50 ms (Varying)
SCH Time	80 ms, 75 ms, 70 ms, 50 ms (Varying)
Physical Layer Properties (Vehicle)	
Standard	IEEE802.11p
Transmitter Power	100 mW
Antenna Gain	1 dBi
Antenna Height	1 m
Bandwidth	10 MHz

Note that the packet trace is enabled under the Configure Reports tab, and the mobility log is enabled under the Network Logs present in the plots on the right side. This allows for the recording of data traffic flow and vehicular movement.

Results When analyzing delay, we change the generation rate such that it is below the saturation capacity of the network. If this were not so, then queuing delay would blow-up at (and post) saturation.

Table 4-6: *We see that as the CCH interval increases, the delay for BSM application decreases. Similarly, as the SCH interval decreases the delay for CBR application increases.*

App	App Type	Gen. Rate (Mbps)	CCH 20ms SCH 80ms	CCH 25ms SCH 75ms	CCH 30ms SCH 70ms	CCH 50ms SCH 50ms
			Delay (Micro sec)			
BSM 1	Broadcast	0.025	38298.82002	33567.59002	29364.14199	15288.49474
BSM 2	Broadcast	0.025	41113.5348	34376.05357	29509.69213	15465.28051
BSM 3	Broadcast	0.025	43021.08925	34569.64036	29879.04223	15326.54219
BSM 4	Broadcast	0.025	38675.9637	34123.20029	30135.26103	15669.02365
Average Delay			40277.35	34159.12	29722.03	15437.33
CBR 1	Unicast	1	9911877.311	10231477.86	46601863.3	46930314.68
CBR 2	Unicast	1	24811785.51	26206924.26	31849109.01	11474867.33
CBR 3	Unicast	1	4244775.494	4518140.717	10246485.37	35513878.28
CBR 4	Unicast	1	47748603.46	48624907.15	3994697.125	4828944.515
Average Delay			21679260.45	22395362.5	23173038.7	24687001.2

Observations

1. CCH Delay has 3 components (a) waiting time where the packet is waiting for the SCH to complete (b) Medium access time and (c) Transmission time.
2. The mathematical analysis of delay is complex. It involves evaluating two difficult components (i) CCH packet waiting time while SCH packet is served and vice versa, and (ii) medium access time. We leave the mathematical analysis to interested researchers, and restrict ourselves to stating that the trends are as expected i.e., increasing CCH time (and reducing SCH time) reduces the CCH delay (and increases SCH delay).

4.3.6 Part 3: Collision count with increasing generation rate

Open NetSim and Select Examples ¿ VANETs ¿ Throughput, delay and collisions with SCH and CCH time division ¿ Collision count with increasing generation rate then click on the tile in the middle panel to load the example as shown in below screenshot.

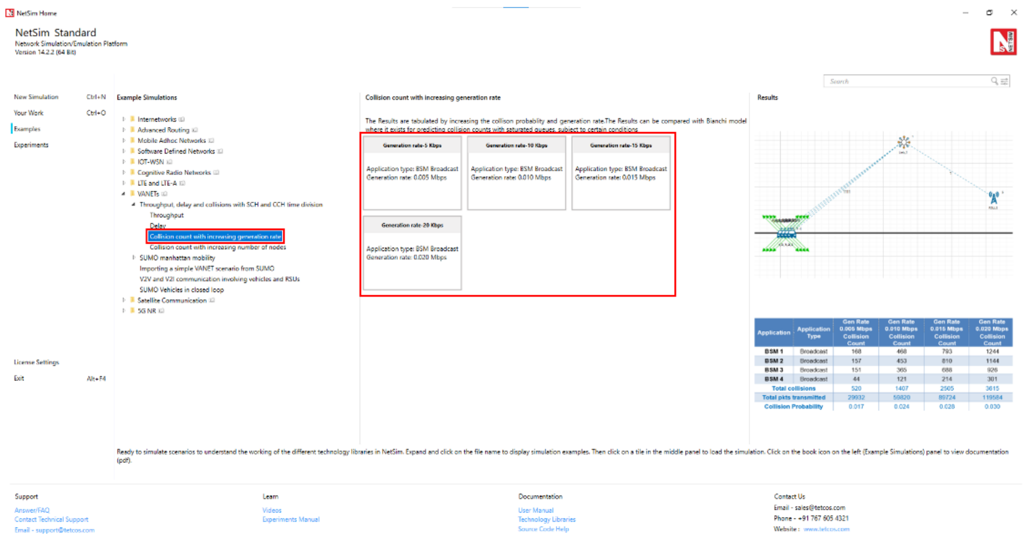


Figure 4-17: *List of scenarios for the example of Collision count with increasing generation rate*

The following network diagram illustrates what the NetSim UI displays when you open the example configuration file as shown in below figure.

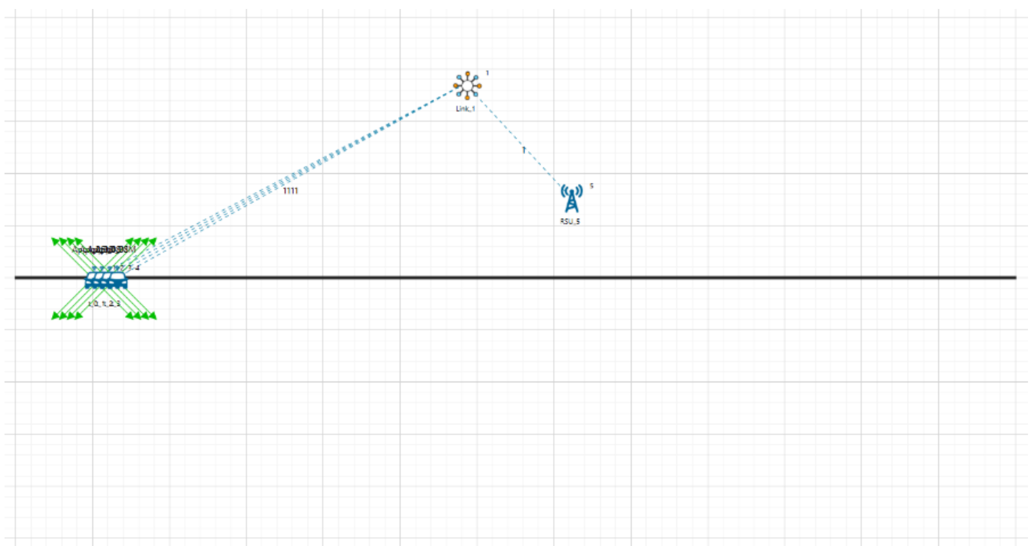


Figure 4-18: *Network setup for studying the Collision count with increasing generation rate*

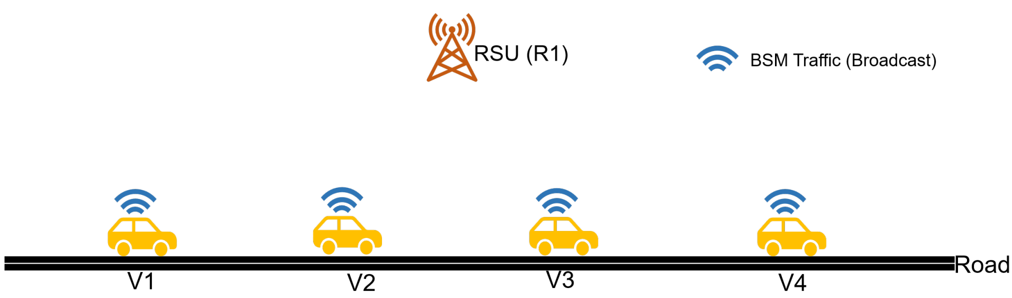


Figure 4-19: *The scenario layout remains the same, however we change the application settings. In this example we only have 4 BSM applications. There are no CBR applications.*

Settings done for the Experiment:

Table 4-7: *Application, link and Physical layer Properties*

Properties	**Properties**
Application Properties	
App Type	BSM
Application Method	Broadcast
Packet Size	20 (Bytes)
Inter Arrival Time	32000 (μs), 16000 (μs), 10666.67 (μs), 8000 (μs) (Varying)
Link Properties	
Channel Characteristics	No pathloss
Datalink Properties	
CCH Time	20 ms
SCH Time	80 ms
Physical Layer Properties (Vehicle)	
Standard	IEEE802.11p
Transmitter Power	100 mW
Antenna Gain	1 dBi
Antenna Height	1 m
Bandwidth	10 MHz

Note that the packet trace is enabled under the Configure Reports tab, and the mobility log is enabled under the Network Logs present in the plots on the right side. This allows for the recording of data traffic flow and vehicular movement.

Results The application generation rates are mentioned in Row 1 (shaded grey).

Table 4-8: *Comparison of Collision count of BSM applications with changing generation rate*

App	**App Type**	**Gen Rate 0.005 Mbps**	**Gen Rate 0.010 Mbps**	**Gen Rate 0.015 Mbps**	**Gen Rate 0.020 Mbps**
		Collision Count	**Collision Count**	**Collision Count**	**Collision Count**
BSM 1	Broadcast	168	468	793	1244
BSM 2	Broadcast	157	453	810	1144
BSM 3	Broadcast	151	365	688	926
BSM 4	Broadcast	44	121	214	301
Total collisions		520	1407	2505	3615
Total pkts transmitted		29932	59820	89724	119584
Collision Probability		0.017	0.024	0.028	0.030

The Collision probability is the ratio between Collision count to total number of packets transmitted:

$$\text{Collision probability} = \frac{\text{Collision count}}{\text{packets transmitted}}$$

To find the Collision count of each individual application:

- Click on Packet trace under traces present in the results dashboard window (Please make sure the packet trace is enabled before running the simulation)

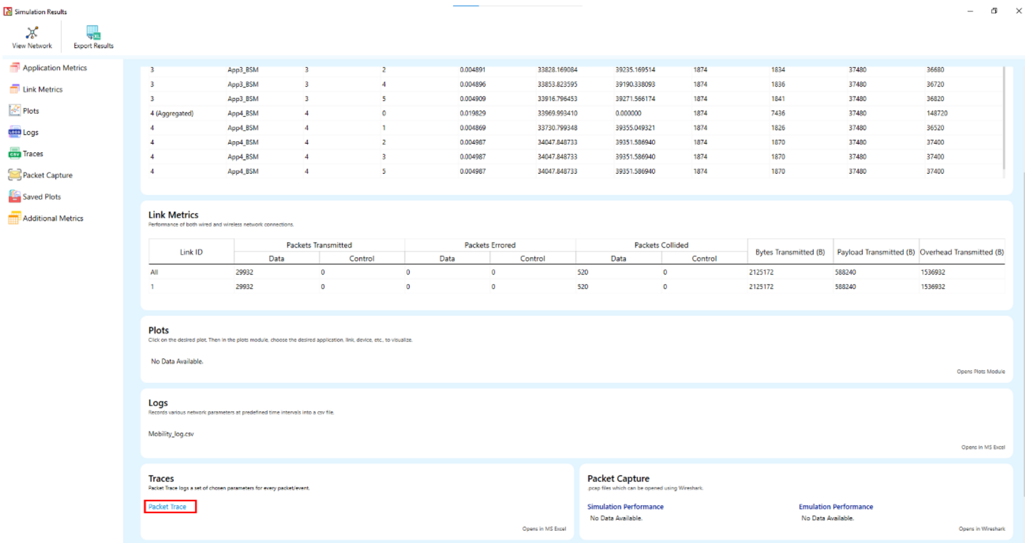


Figure 4-20: *Results dashboard window*

- Total no of collision counts, and packets transmitted can be viewed in the link metrics table over results dashboard.
- In packet trace, filter the control packet type / App Name to App1_BSM to find the individual collision count.
- Along with that filter the packet status field to collisions to view the collisions of that application (APP1 BSM).

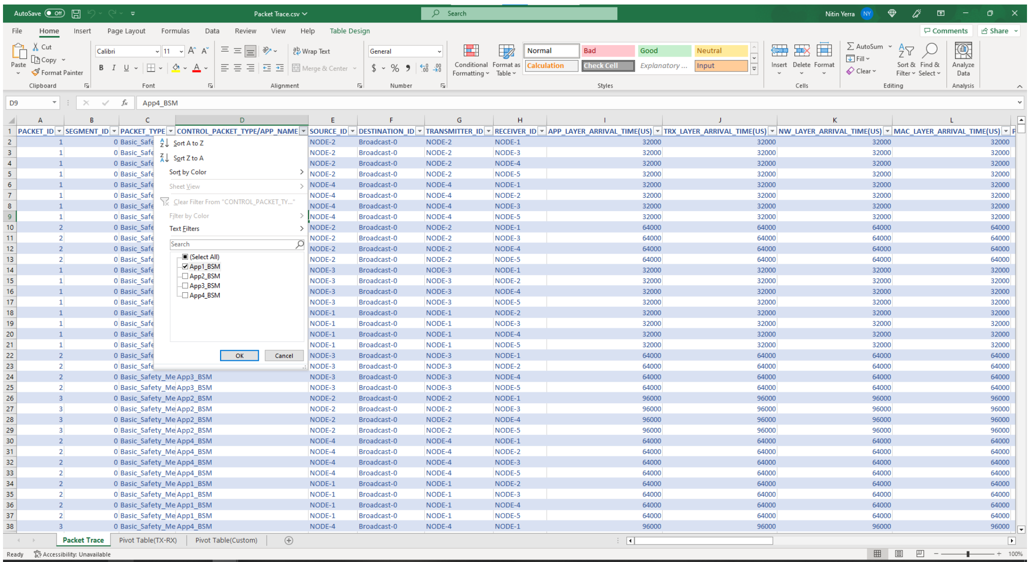


Figure 4-21: *Packet trace which depicts filtering of applications*

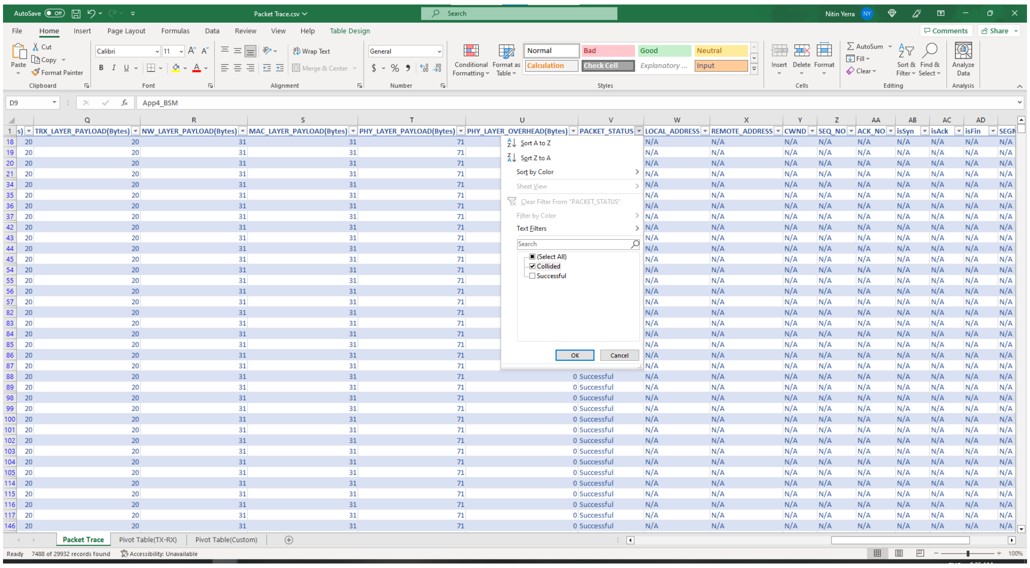


Figure 4-22: *Packet trace that depicts filtering of packet status of each application*

- After applying the filters, the total collision count of APP1 BSM applications can be viewed.

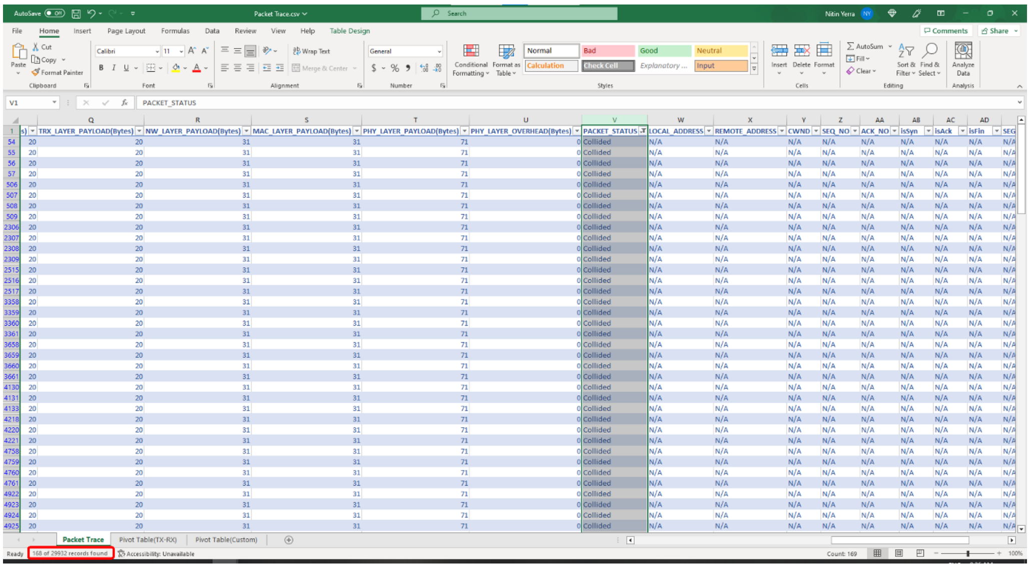


Figure 4-23: *Packet trace*

Same process can be done for all the remaining applications of the network.

Observations

- Saturation throughput is about 0.25 Mbps per app or 1 Mbps total. Note the generation rate is below the saturation capacity of the network.
- We see collision probability increases as generation rate increases.
- To the best of our knowledge the mathematical modelling of collisions with non-saturated queues is an open problem. The Bianchi model exists for predicting collision counts with saturated queues, subject to certain conditions.

4.3.7 Part 4: Collisions count with increasing number of nodes

Open NetSim and Select Examples ¿ VANETs ¿ Throughput, delay and collisions with SCH and CCH time division ¿ Collisions count with increasing number of nodes then click on the tile in the middle panel to load the example as shown in below screenshot.

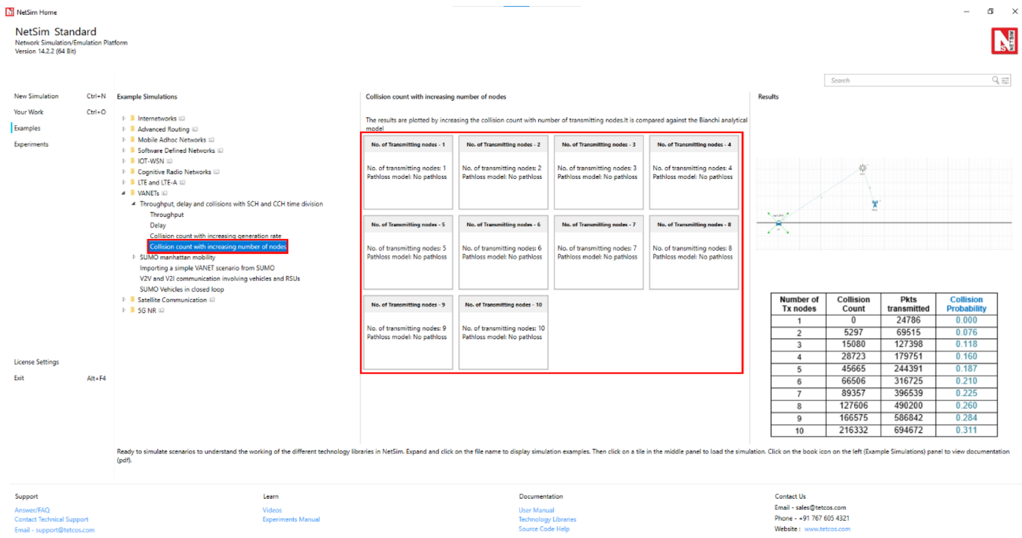


Figure 4-24: *List of scenarios for the example of Collisions count with increasing number of nodes*

The following network diagram illustrates what the NetSim UI displays when you open the example configuration file as shown in below figure.

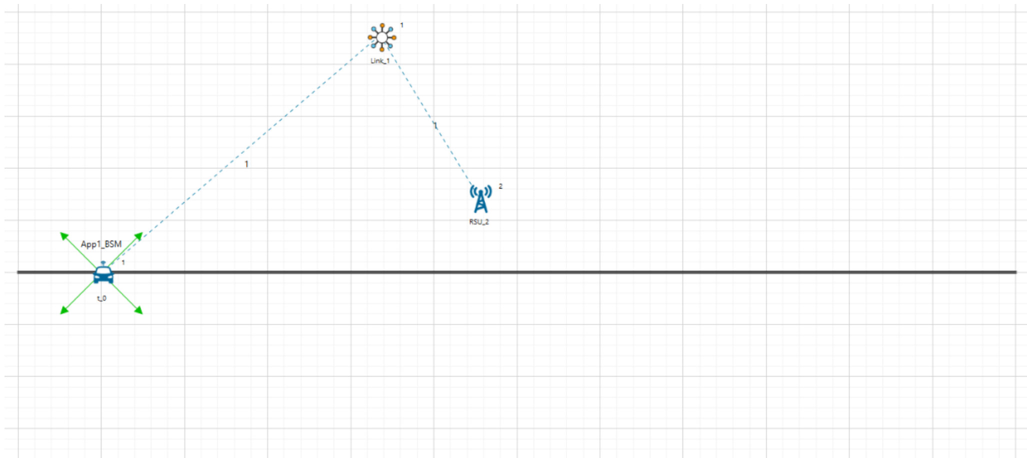


Figure 4-25: *Network set up for studying the Collisions count with increasing number of nodes*

This scenario has 10 vehicles in a line on a road. The vehicles transmit power $P_t = 20$ dBm, Carrier sense threshold $CS_{th} = -85$ dBm, and we assumed log distance pathloss with $\eta = 2.5$. The received power between nodes with maximum separation, $d = 100$, is:

$$P_r = 20 - 47.88 - 10 \times 2.5 \times \log(100) = -77.88 \text{ dBm}$$

Based on the formula:

$$Rx_{\text{Power}} = Tx_{\text{Power}} + G_t + G_R - PL_{d0} - 10\log(D^{\eta})$$

For detailed understanding of the receive power calculations refer to Propagation model:

https://tetcos.com/downloads/v14.1/Propagation-Models.pdf

Since $P_r > CS_{th}$ all nodes can hear one another which means that they are all within CS Range.

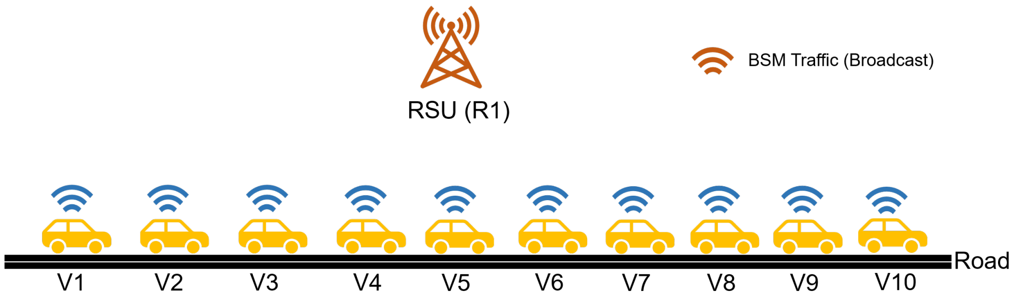


Figure 4-26: *Illustration of the VANET scenario under study. The network comprises of 10 vehicles and 1 roadside unit. Each vehicle transmits one application (i) a BSM broadcast application that is sent to all other devices (vehicles plus RSU) within range. In this study we are increasing the Tx nodes from 1–10*

Settings done for the Experiment:

Table 4-9: *Application, Link and Physical layer Properties*

Properties	Properties
Application Properties	
App Type	BSM
Application Method	BROADCAST
Packet Size	20 (Bytes)
Inter Arrival Time	320 (μs)
Link Properties	
Channel Characteristics	No Pathloss
Data Link Properties	
CCH Time	20 ms
SCH Time	80 ms
Physical Layer Properties (Vehicle)	
Standard	IEEE802.11p
Transmitter Power	100 mW
Antenna Gain	1 dBi
Antenna Height	1 m
Bandwidth	10 MHz

Note that the packet trace is enabled under the Configure Reports tab, and the mobility log is enabled under the Network Logs present in the plots on the right side. This allows for the recording of data traffic flow and vehicular movement.

Results

Table 4-10: *Collision probability comparison with change in number of transmitting nodes*

Number of Tx nodes	Collision Count	Pkts transmitted	Collision Probability
1	0	31261	0.000
2	0	75726	0.000
3	1731	127041	0.014
4	7142	187016	0.038
5	16724	252210	0.066
6	32045	325308	0.099
7	53946	405678	0.133
8	86203	499376	0.173
9	126970	596448	0.213
10	176435	704510	0.250

The total number of collision counts and packets transmitted can be viewed in link metrics window of results dashboard.

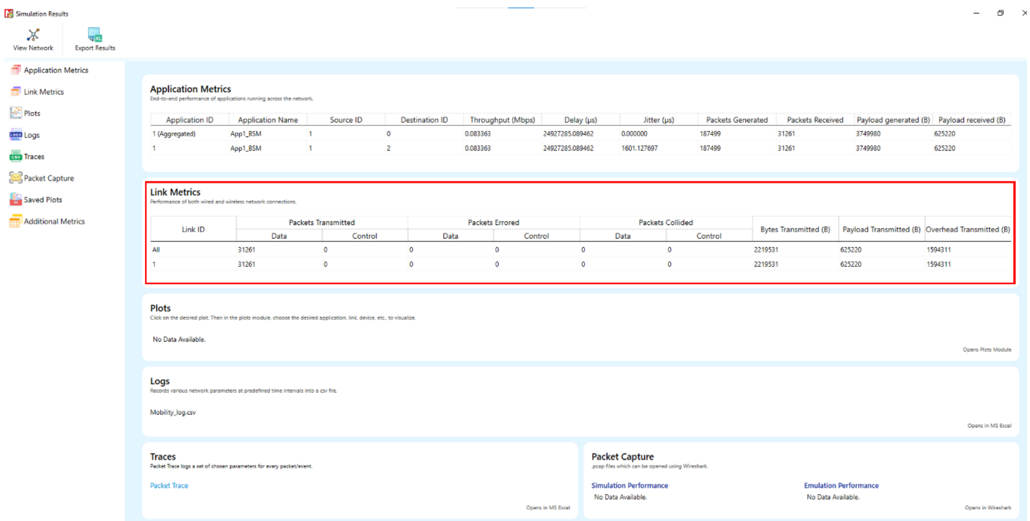


Figure 4-27: *Results Dashboard window*

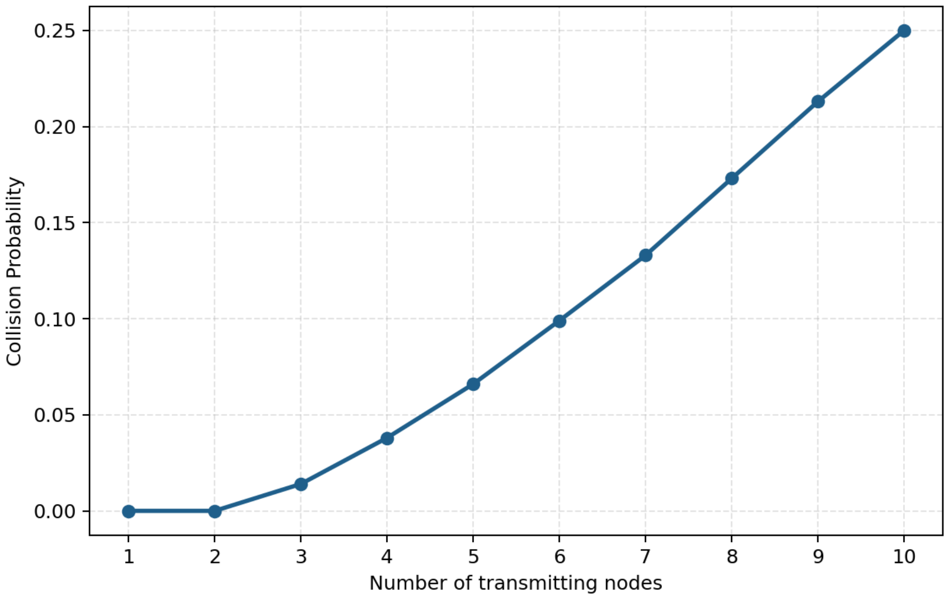


Figure 4-28: *Collision probability vs. number of transmitting nodes*

The Collision probability is the ratio between Collision count to total number of packets transmitted:

$$\text{Collision probability} = \frac{\text{Collision count}}{\text{packets transmitted}}$$

Observations

- We see collision count increasing with number of transmitting nodes.
- This can be compared against the Bianchi analytical model.

4.4 SUMO Manhattan Mobility with Single and Multi-hop Communication

Introduction

The Manhattan mobility in SUMO features a grid topology as shown below. It is composed of a number of horizontal and vertical streets. Each street has two lanes for each direction (North and South direction for vertical streets, East and West for horizontal streets). The mobile node is allowed to move along the grid of horizontal and vertical streets. At an intersection of a horizontal and a vertical street, the mobile node can turn left, right or go straight with certain probability.

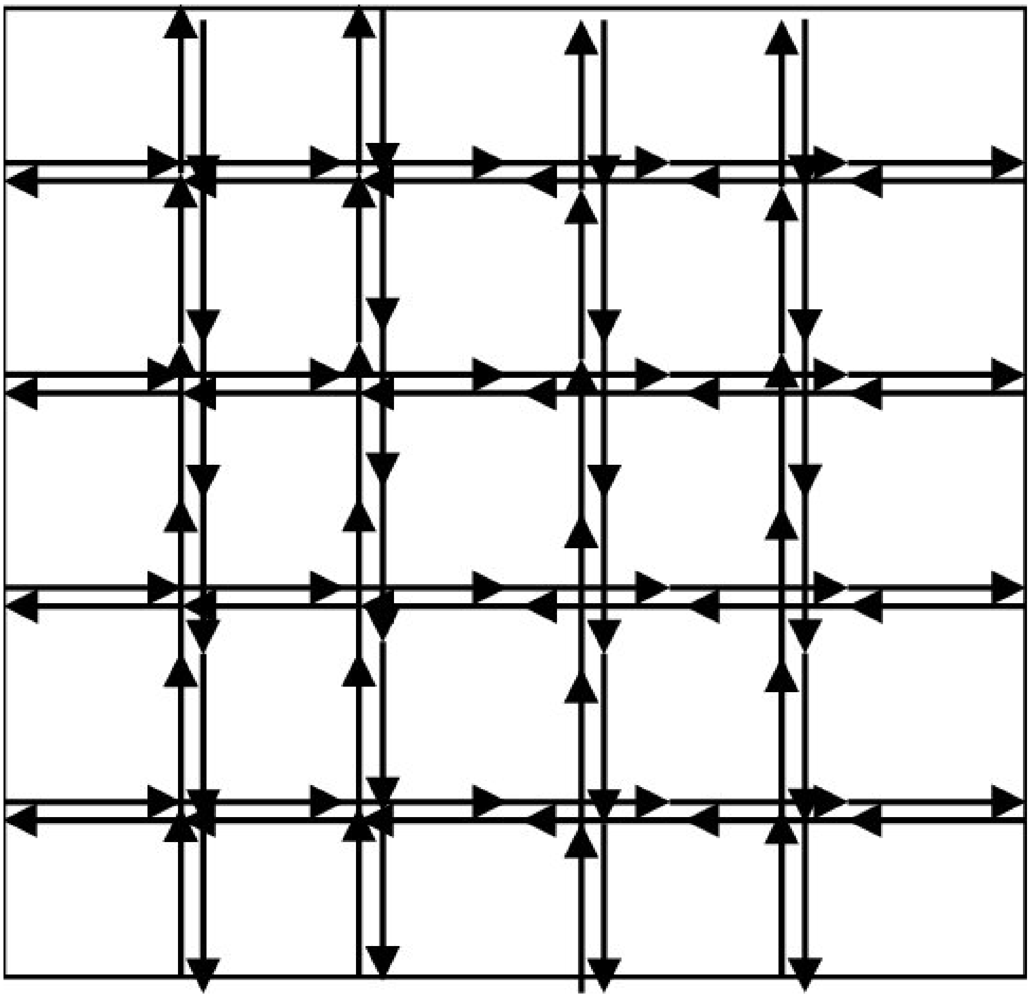


Figure 4-29: *Manhattan mobility in SUMO features a grid topology*

Case 1: Manhattan mobility Single-hop RSU to vehicles

Objective

To create, using SUMO, a Manhattan Road network in which vehicles drive randomly, and to have a Roadside unit (RSU) which sends safety messages continuously to vehicles. The network performance is analyzed for different environments each having different RF channel characteristics.

Procedure

Open NetSim and Select Examples ¿ VANETs ¿ SUMO Manhattan mobility ¿ Single hop communication then click on the tile in the middle panel to load the example as shown in below screenshot.

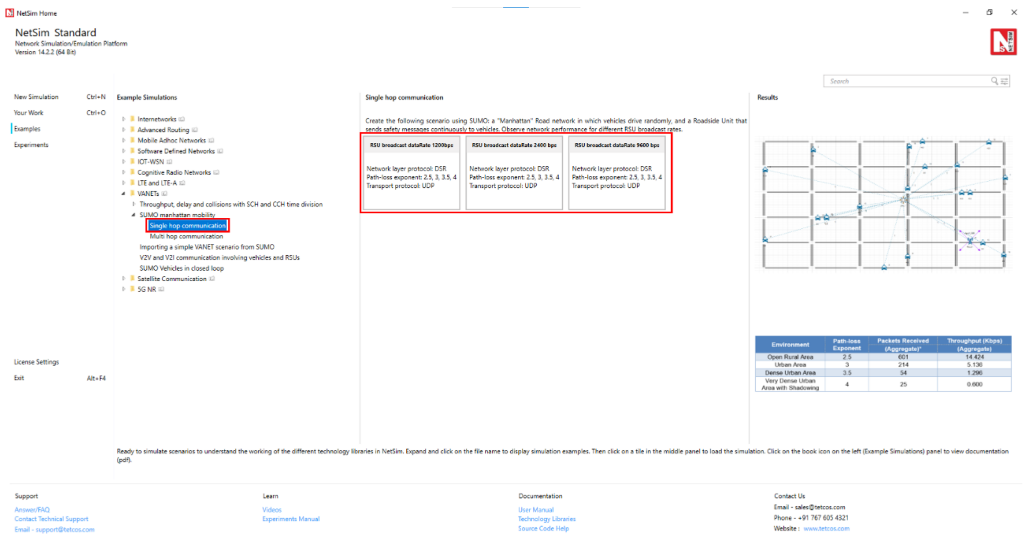


Figure 4-30: *List of scenarios for the example of Single hop communication*

The NetSim UI would display as shown below.

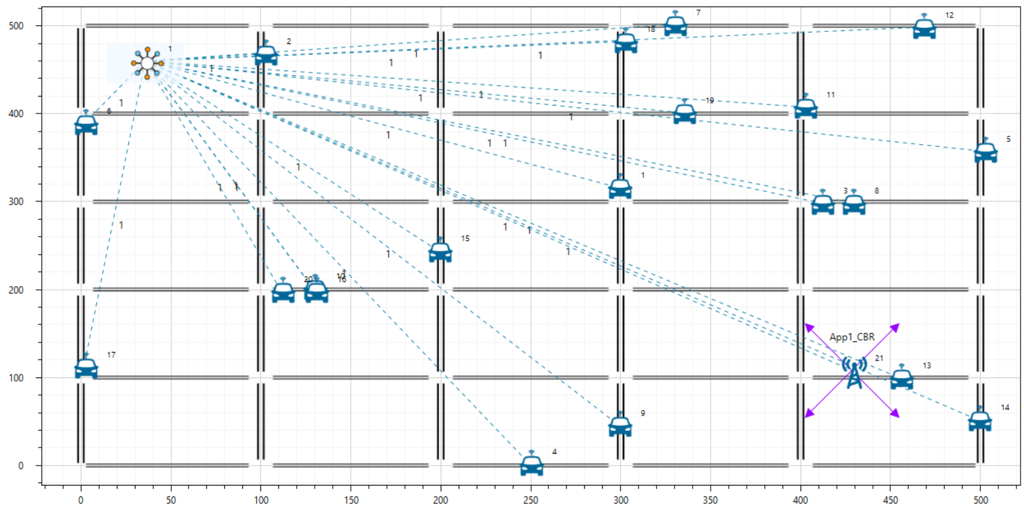


Figure 4-31: *Network setup for studying the Single hop communication*

Settings done for this sample experiment.

1. Configure CBR Applications (Broadcast application) set as below properties:

Table 4-11: *CBR Applications Settings*

App Method	App Type	App Name	Source ID	Destination ID	Pkt Size (B)	IAT (μs)
Broadcast	CBR	APP 1 CBR Broadcast	21	Broadcast to all 20 vehicles	300	2,000,000

2. Transport protocol set as UDP in application Configuring window.
3. Click on ad hoc link/wireless link, expand the right-side property panel and set the properties as follows:

Table 4-12: *Wireless link properties*

Channel characteristics	Pathloss Model	Pathloss Exponent
Pathloss Only	Log distance	2.5

4. Co-ordinates of RSU are set as X = 428.61, and Y = 108.06.
5. Set transmitter power to 1000mW under Interface 1(wireless) ¿ Physical layer properties of vehicles and RSU.
6. The packet trace is enabled under configure reports tab and the mobility log is enabled under the network logs present in the plots on the right side, allowing the recording of data traffic flow and the vehicular movement.
7. Increase the pathloss exponent (in the order 2.5, 3, 3.5, 4) and note down the aggregate throughput and packets received for different application generation rates.
8. After running the simulation, Packet Trace can be used to visualize packet flow along with packet information and Mobility log can be used to record vehicle mobility. Time varying throughput plot can be opened from the Results window.

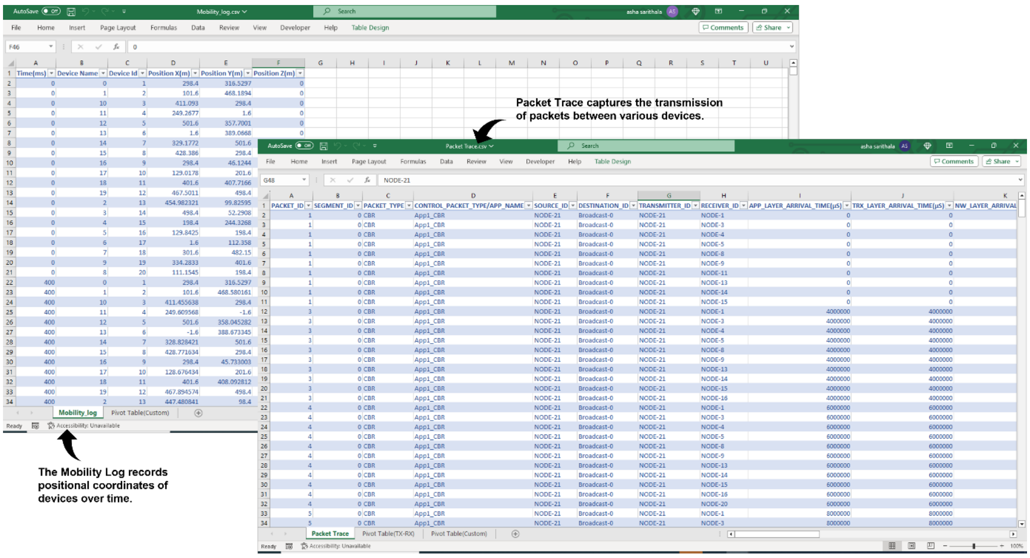


Figure 4-32: *Packet Trace and Mobility log window*

9. In SUMO GUI, you can see that vehicles choose random directions when they reach a junction in the Manhattan grid network.

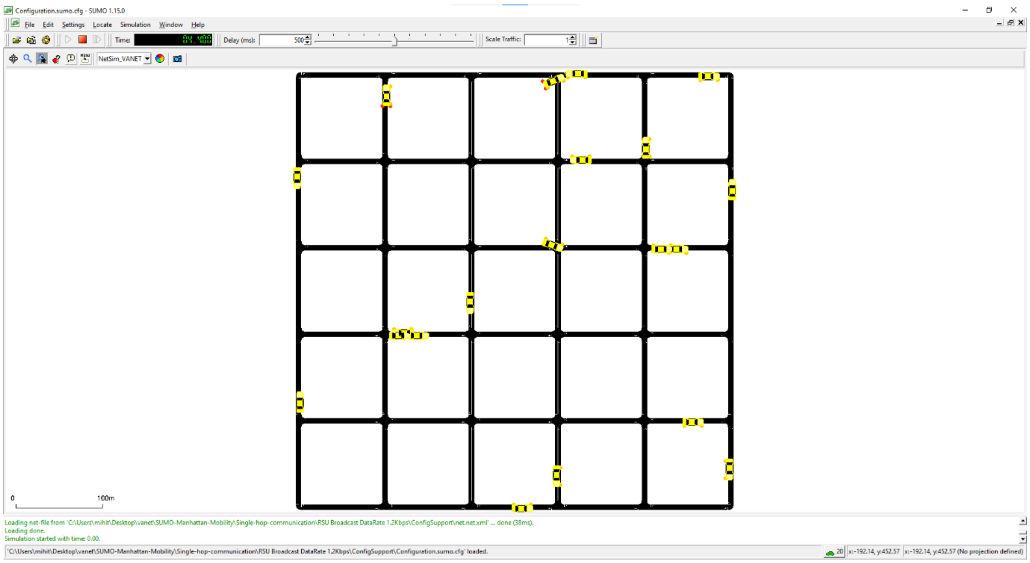


Figure 4-33: *Vehicles movement in SUMO-GUI window*

Results and Observations

For sample RSU Broadcast Data Rate = 1.2 Kbps (Packet size = 300 bytes, IAT = 2,000,000μs. This means packets of size 300 Bytes are sent every 2 seconds).

Table 4-13: *Results Comparison for RSU Broadcast Data Rate = 1.2 Kbps*

Environment	Path-loss Exponent	Packets Received (Aggregate)	Throughput (Kbps) (Aggregate)
Open Rural Area	2.5	980	23.52
Urban Area	3	511	12.264
Dense Urban Area	3.5	209	5.016
Very Dense Urban Area with Shadowing	4	57	1.368

* Aggregate is the sum of the packet/throughputs obtained by all applications.

For sample RSU Broadcast Data Rate = 2.4 Kbps (Packet size = 300 Bytes, IAT = 1,000,000μs or 1 second. This means packets of size 300 Bytes are sent every second).

Table 4-14: *Results Comparison for RSU Broadcast Data Rate = 2.4 Kbps*

Environment	Path-loss Exponent	Packets Received (Aggregate)	Throughput (Kbps) (Aggregate)
Open Rural Area	2.5	1960	47.04
Urban Area	3	1022	24.528
Dense Urban Area	3.5	413	9.912
Very Dense Urban Area with Shadowing	4	114	2.736

For sample RSU Broadcast Data Rate = 9.6 Kbps (Packet size = 300 Bytes, IAT = 250,000μs or 0.25 seconds. This means four packets of size 300 Bytes are sent every 0.25 second).

Table 4-15: *Results Comparison for RSU Broadcast Data Rate = 9.6 Kbps*

Environment	Path-loss Exponent	Packets Received (Aggregate)	Throughput (Kbps) (Aggregate)
Open Rural Area	2.5	7920	190.08
Urban Area	3	4107	98.568
Dense Urban Area	3.5	1659	39.816
Very Dense Urban Area with Shadowing	4	461	11.064

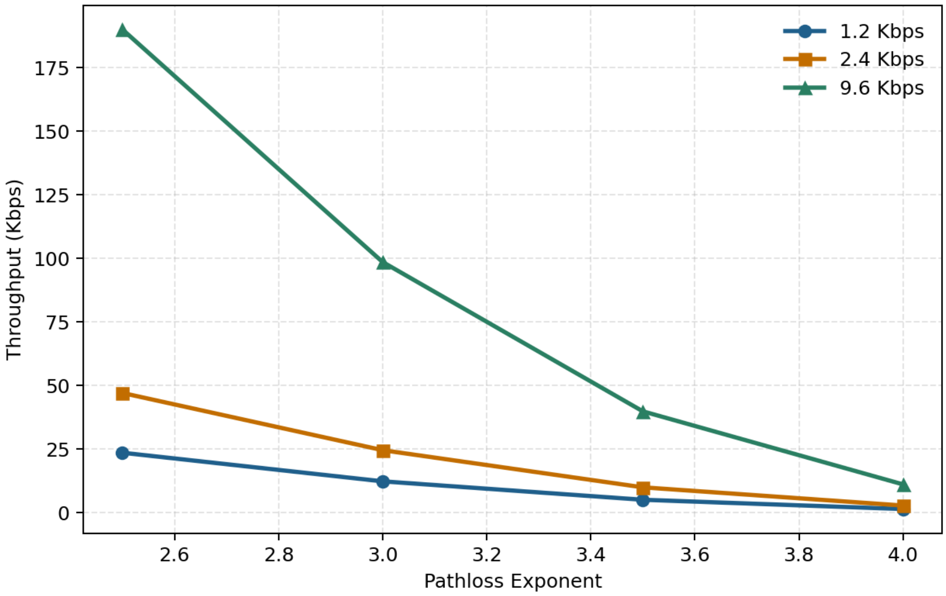


Figure 4-34: *Plot of Throughput vs. Pathloss Exponent for different RSU broadcast for different DR (Data Rates)*

Case 2: Manhattan mobility Multi-hop Vehicles to RSU

Objective

To create, using SUMO, a Manhattan Road network in which vehicles drive randomly, and to have a Roadside unit (RSU) to which vehicles continuously send unicast traffic via multi-hop (hopping via other vehicles if the RSU is beyond communication range). The network performance is analyzed for different vehicle counts.

Procedure

Open NetSim and Select Examples ¿ VANETs ¿ SUMO Manhattan mobility ¿ Multi hop communication then click on the tile in the middle panel to load the example.

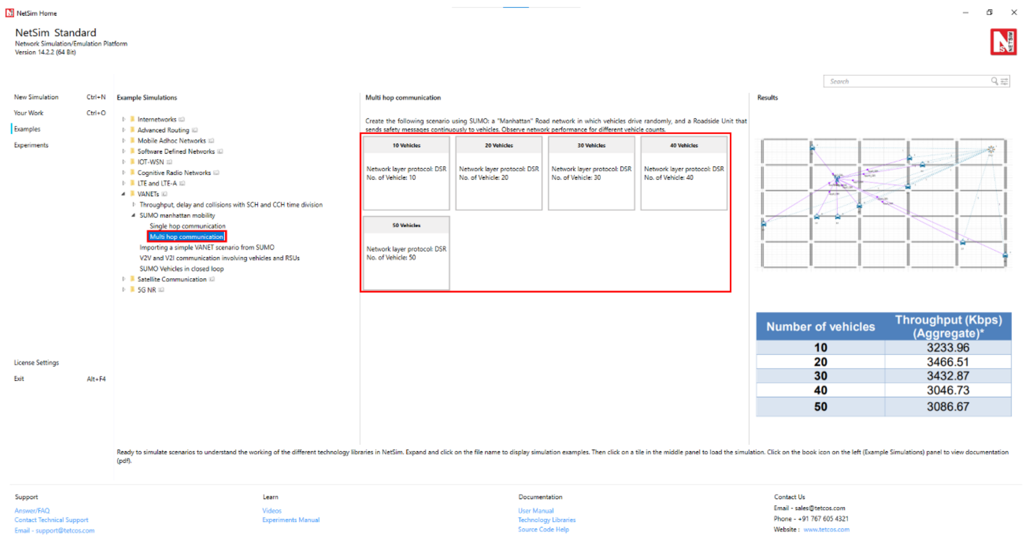


Figure 4-35: *List of scenarios for the example of Multi hop communication.*

The NetSim UI would display as shown below.

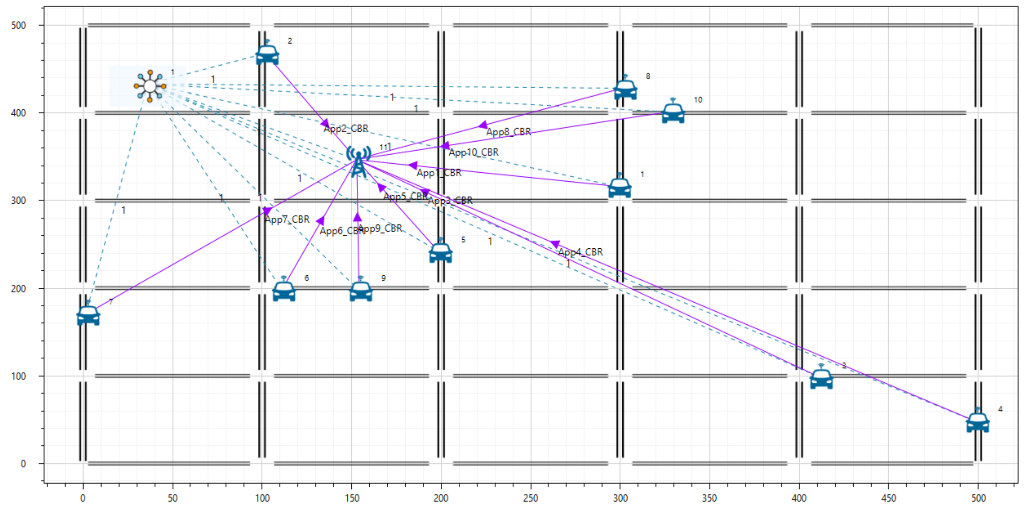


Figure 4-36: *Network set up for studying the Multi hop communication*

Settings done for this sample experiment.

1. Configure CBR Application between the nodes as shown in the table and, click on application, expand the right-side property panel and set the application properties below.

Table 4-16: *CBR Applications settings*

App Method	App Type	Source Id	Destination Id	Pkt Size (B)	IAT (μs)
Unicast	CBR	(All vehicles)	RSU	1460	20,000

2. In Vehicle General Properties, under SUMO file Configuration.sumo.cfg file was selected from the Docs folder of NetSim Install Directory `<C:\Program Files\NetSim Standard\Docs\Sample-Configuration\VANET\SUMO`

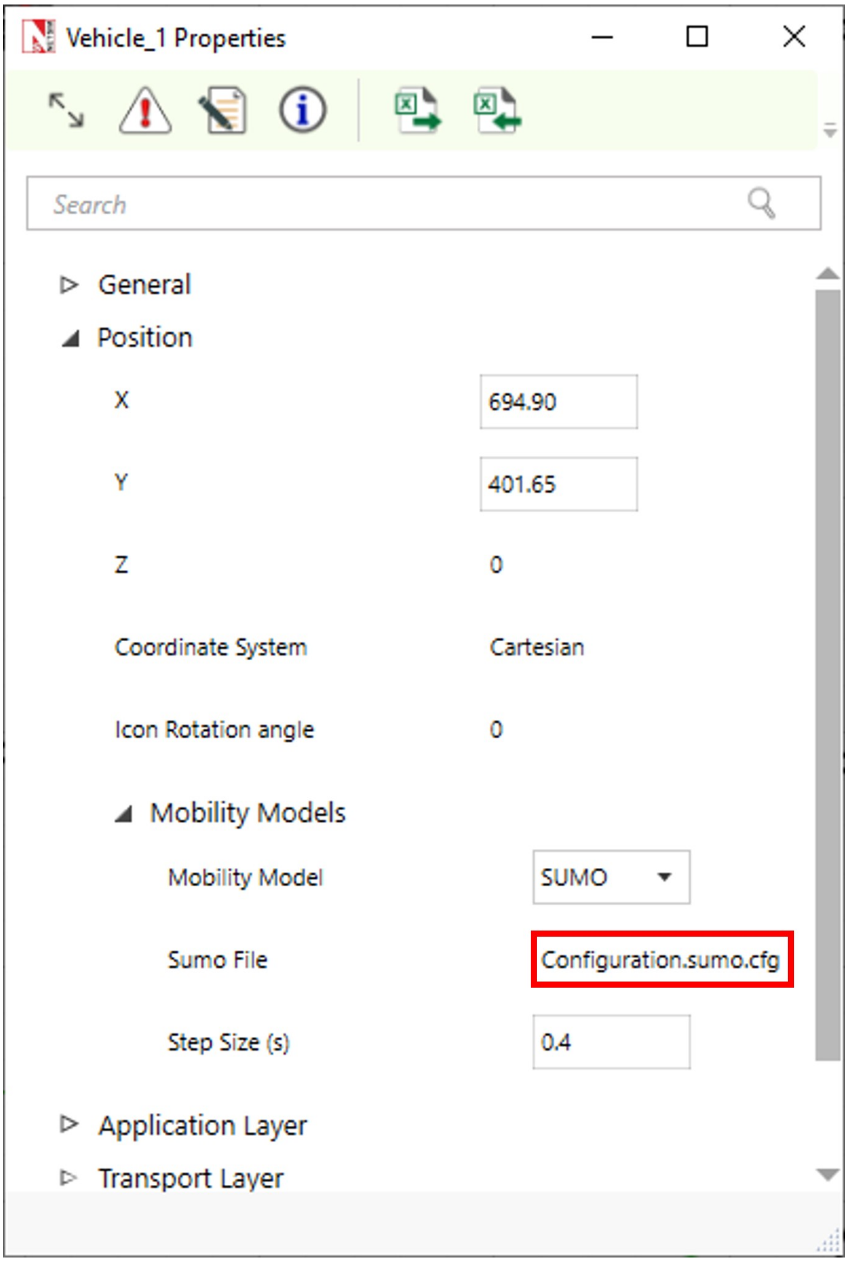


Figure 4-37: *General Properties window*

3. Transport protocol set as UDP in application Configuration window.
4. Click on Adhoc link / Wireless link and expand the right-side property panel and set Pathloss as No Pathloss.
5. RSU is dropped randomly as it is No Pathloss.
6. Click on the Vehicles and RSU and expand the right-side property panel and set Network layer routing protocol as DSR.
7. Set transmitter power to 1000mW under Interface 1(wireless) ¿ Physical layer properties of Vehicles and RSU.
8. Mobility log is enabled under the network logs present in the plots on the right side, allowing the vehicular movement.
9. Run the simulation.
10. Increase the number of vehicles in the order 10, 20, 30 etc. and note down the aggregate throughput.

Result:

Table 4-17: *Results Comparison*

Number of vehicles	Throughput (Kbps) (Aggregate)*
10	1794.747
20	1575.164
30	1675.729
40	2579.062
50	1923.464

*Aggregate is the sum of the packet/throughputs obtained by all applications.

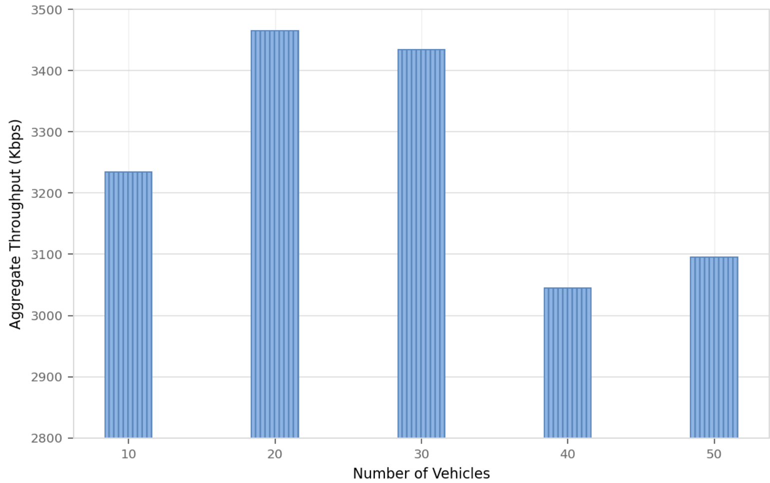


Figure 4-38: *Aggregate Throughput vs. Number of Vehicles*

4.5 SUMO Interfacing with vehicles moving in a closed loop

Open NetSim and Select Examples ¿ VANETs ¿ SUMO Vehicles in closed loop then click on the tile in the middle panel to load the example as shown in below figure.

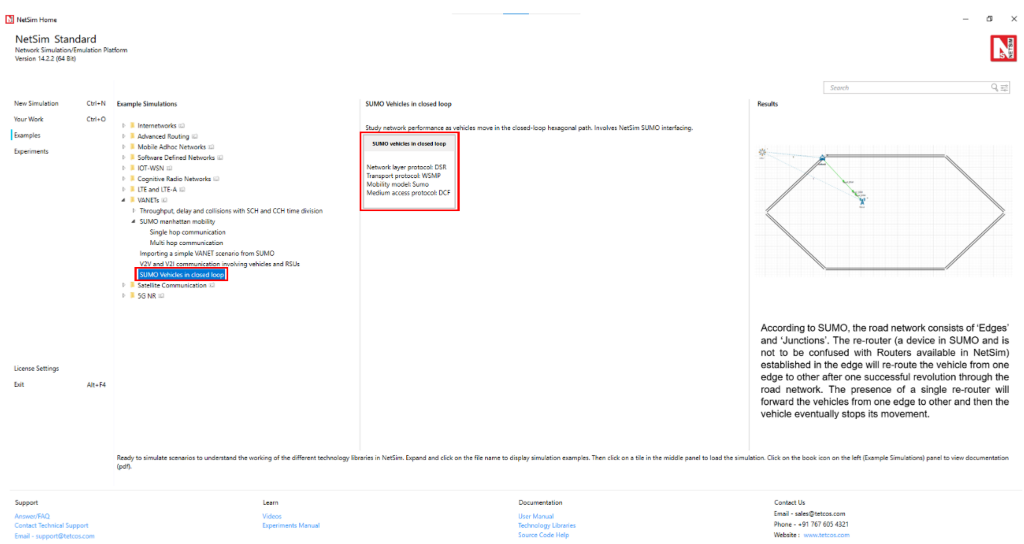


Figure 4-39: *List of scenarios for the example of SUMO Vehicles in closed loop*

The NetSim UI would display as shown below.

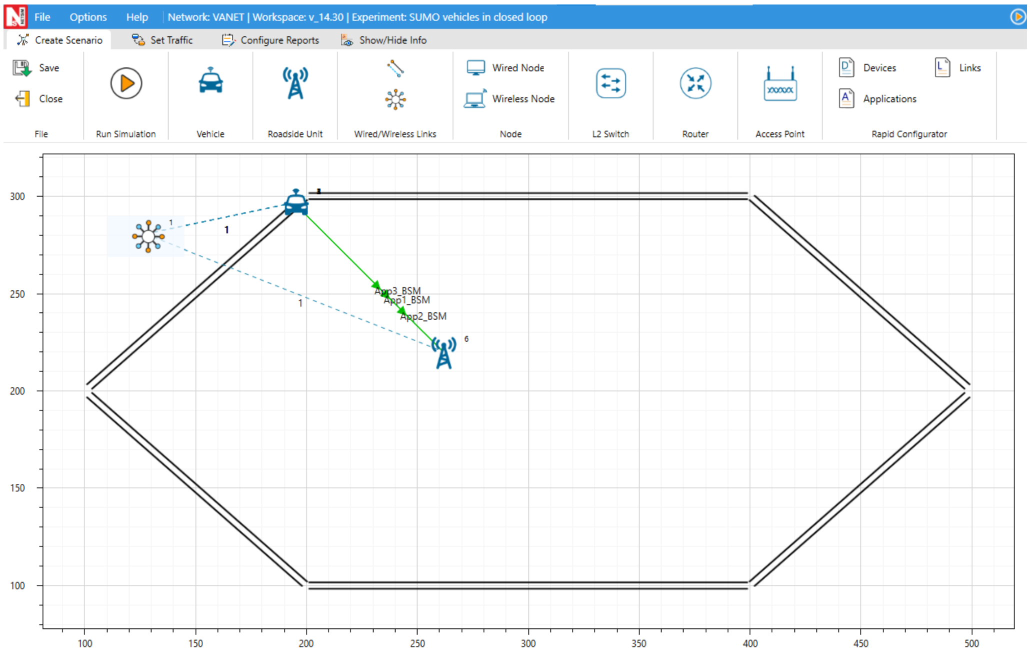


Figure 4-40: *Network set up for studying the SUMO Vehicles in closed loop*

Settings done for this sample experiment:

1. Configure BSM (Basic Safety Message) Application between the nodes as shown in the table and, click on application, expand the right-side property panel and set the application properties below.

Table 4-18: *BSM Applications settings*

APP_ID	Source ID	Destination ID	Packet Size (Bytes)	Inter-Arrival Time (μs)
APP_1_BSM	1	6 (RSU)	20	20,000
APP_2_BSM	2	6 (RSU)	20	20,000
APP_3_BSM	3	6 (RSU)	20	20,000

2. Transport protocol set as WSMP for all applications in application window.
3. In Vehicle Position properties, under SUMO file Configuration.sumo.cfg file was selected from the Docs folder of NetSim Install Directory <C:\Program Files\NetSim Standard\Docs\Sample-Configuration\VANET\SUMO

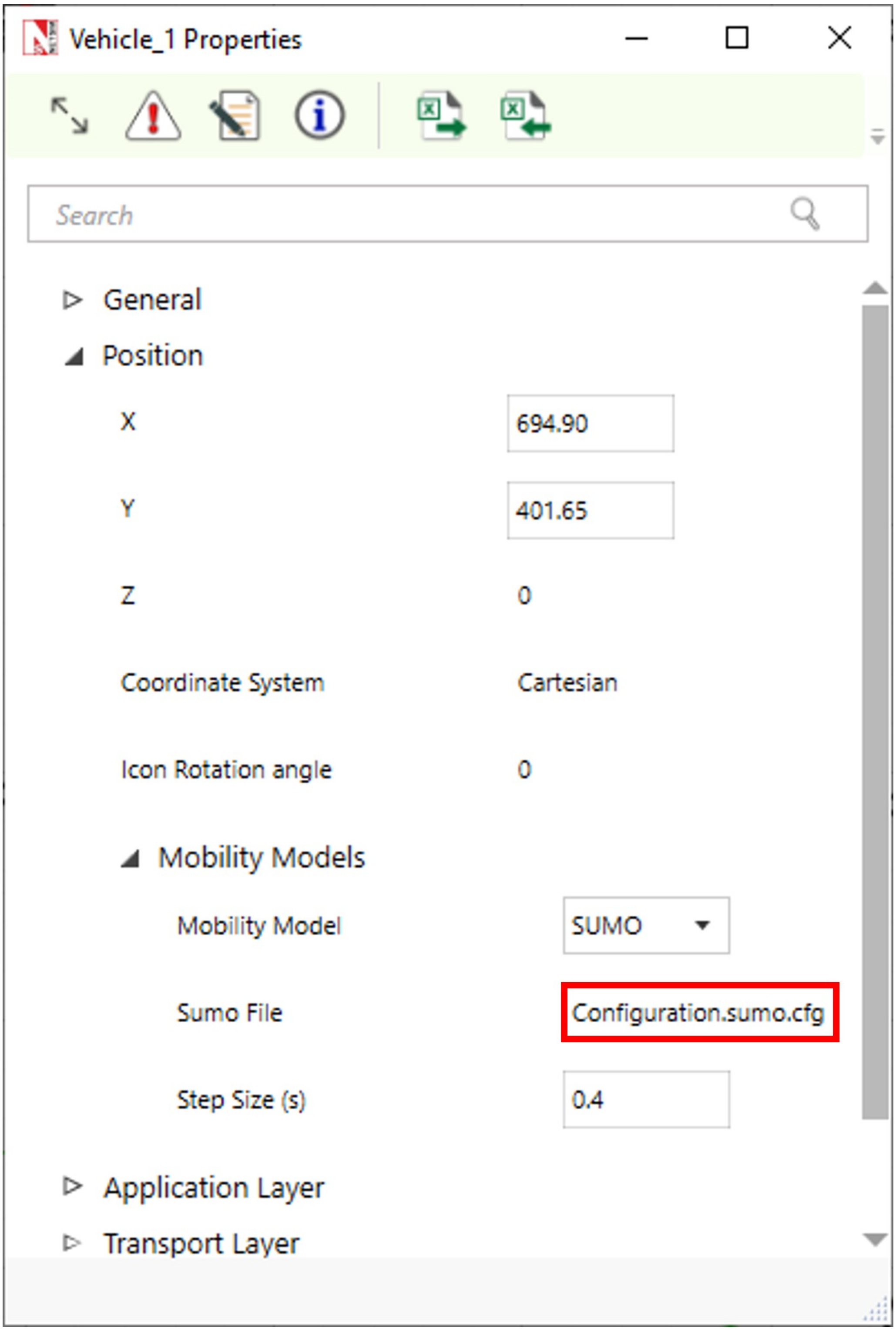


Figure 4-41: *General Properties window*

4. Click on Adhoc link/Wireless link and expand the right-side property panel and set Pathloss as No pathloss.
5. RSU is dropped randomly as it is set to No pathloss.
6. Click on vehicles and RSU, then expand the right-side property panel. Go to Interface(wireless) ¿ Datalink layer Properties and set DCF as the Medium Access Protocol.
7. Set transmitter power to 1000mW under Interface 1(wireless) ¿ physical layer properties of vehicles and RSU.

8. Note that the packet trace is enabled under the Configure reports tab, and the mobility log is enabled under the Network Logs present in the plots on the right side. This allows for the recording of data traffic flow and vehicular movement and run simulation.
9. After running the simulation, Packet trace can be used to visualize packet flow along with packet information and mobility log can be used to record vehicle mobility. Time varying throughput plot can be opened from the Results window.

Result:

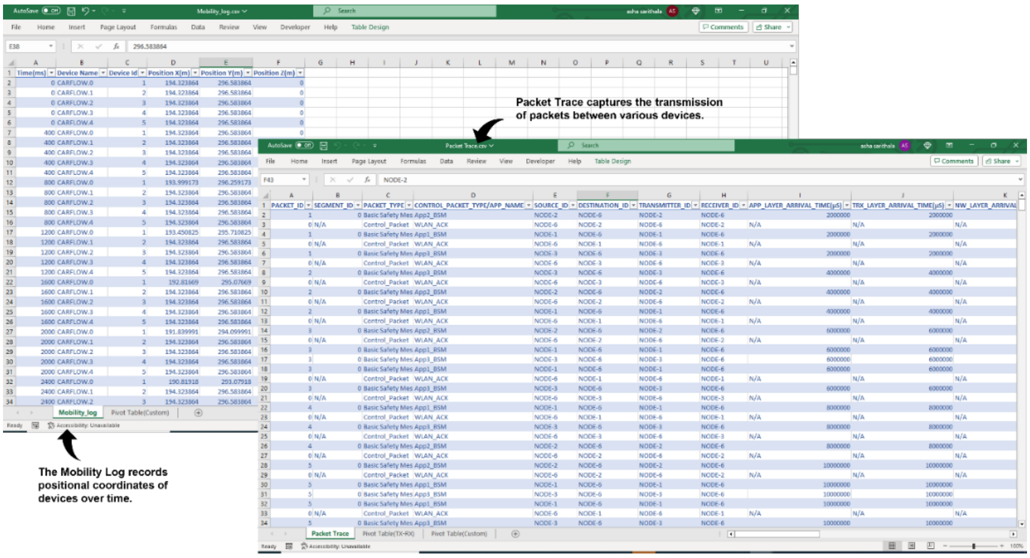


Figure 4-42: *Packet Trace and Mobility log window.*

Same can be observed in SUMO as well:

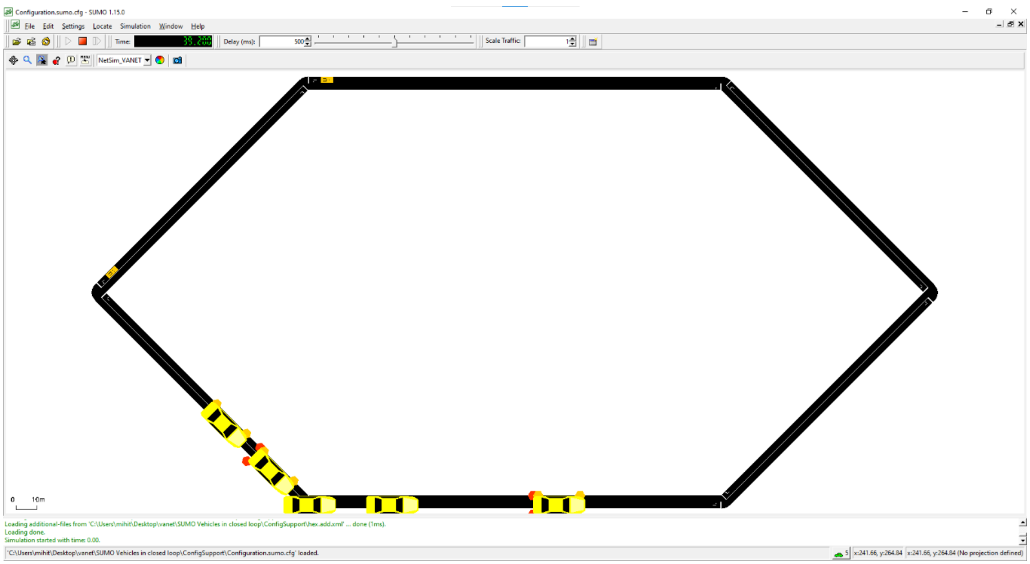


Figure 4-43: *Animation window for Sumo*

According to SUMO, the road network consists of 'Edges' and 'Junctions'. The re-router (a device in SUMO and is not to be confused with Routers available in NetSim) established in the edge will re-route the vehicle from one edge to another after one successful revolution through the road network. The presence of a single re-router will forward the vehicles from one edge to other and then the vehicle eventually stops its movement. Hence, two re-routers have been established in two edges which re-route the vehicle from one edge to other. The above road network consists of six edges in which re-routers are established in the starting and ending edges, which re-routes the vehicles present in the network from starting edge to the finishing edge after one complete revolution through the road or path. As a result, the vehicles will move through the closed loop continuously, until the end time configured in the configuration file.

The RSU configured in the network will allow V2I communication. Per the application configuration a 100 bytes packet is transmitted from vehicle to RSU every 2 seconds. This can also be observed in the packet trace.

5 Reference Documents

- IEEE 802.11p 2010. Wireless Access for Vehicular Environments
- IEEE 1609: Standards for Wireless Access in Vehicular Environment (WAVE)

6 Latest FAQs

1. Up to date FAQs on NetSim's VANETs library is available at
2. https://tetcos.freshdesk.com/support/solutions/folders/14000118424